1966-10

Ocean wave crest and ray refraction in shoaling water by computer

Stouppe, David Edward

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/9599

OCEAN WAVE CREST AND RAY REFRACTION IN SHOALING WATER BY COMPUTER

DAVID EDWARD STOUPPE

517

OCEAN WAVE CREST AND RAY REFRACTION IN SHOALING

WATER BY COMPUTER

by

David Edward Stouppe
Lieutenant, United States Navy
B.S.A.E., Purdue University, 1957

Submitted in partial fulfillment
for the degree of

MASTER OF SCIENCE IN PHYSICAL OCEANOGRAPHY

from the

UNITED STATES NAVAL POSTGRADUATE SCHOOL
October 1966

ABSTRACT

The knowledge of wave refraction is important in many studies. The rapid and relatively easy gaining of this knowledge is made possible by the use of the modern high-speed digital computer. Large numbers of spectral periods and incoming directions are easily investigated, and immediate results are obtained by use of the plot of the wave crest refraction from the computer. This program presents the wave crest refraction pattern of numerous wave ray points rather than the single ray following technique. Its use is valuable in amphibious operation planning, and in other studies where the distribution of wave energy along the shore is desired for the many periods of the wave spectrum.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

1. Introduction

The knowledge of the refraction pattern as an ocean wave approaches the shore is necessary in the determination of the energies of the wave in the surf zone. Wave energy is related in turn to many near shore processes or operations, such as beach erosion, sediment movement, or amphibious landing operations.

While this investigation is slanted toward the needs for amphibious operations, the knowledge and methods used are amenable to any users who desire the pattern of wave crest or refraction, or an estimate of the ratio of the wave height at a point to the deep-water wave height (H/H_0) for other purposes.

Early work on the subject was concerned mostly with hand calculation of the various determining parameters and graphical method for constructing the refraction diagrams [1] [2]. Later, in early 1962, attempts were made to evaluate wave refraction using high-speed digital computers [3]. The most usable results published were those obtained by Harrison and Wilson [4]. While the work of Griswold and Nagle as well as that of Harrison and Wilson gives worthwhile results, both studies are very limited with respect to the ratio of work input to the results obtained. They are concerned with following one ray at a time from deep water to the shoreline. To get a complete refraction diagram many inputs to the computer are required, and hand plotting of the results are necessary.

The procedure outlined in this paper follows a wave crest composed of a number of ray points to the shoreline with immediately available results. No hand plotting or further program input is required for the rays with one period and the same initial deep-water direction. By changing only the wave period or wave direction, the spectrum of periods

and range of angles can be investigated rapidly with little additional time involved.

All of the computer programs for the investigation of wave refraction utilize the same basic procedure: following the wave ray (orthogonal) to the shoreline. The new program uses a depth field as an interpolating surface rather than the velocity field of [3] and [4]. Several interpolation surfaces have been investigated to represent the velocity or depth field used [4]. This program utilizes a quadric surface for interpolation of the depth field.

The present study uses an input of deep-water wave period and wave direction to a computer program for determining wave refraction in a method such as proposed by Munk and Arthur, which employs the wave parameters listed in Appendix IV [5]. An additional subroutine of the program computes the coefficient of refraction, K, and the ratio of wave heights, H/H_0. The latter two values are recorded along with the X and Y coordinates of each point along the wave crest. As a rapid means of viewing the refraction pattern of the wave crest, a graphical output is included which contours every third wave crest computed. Other parameters may be included in the output at the user's discretion.

2. Method.

The first step in the utilization of the program is the construction of a grid of depth values for the desired area. This grid must include starting points in deep water for all rays to be followed, such that the ratio of the depth to the deep-water wave length, d/L_0, is greater than 0.5. Since the program is arranged to follow the waves from deep water to the shore, the grid origin must be in deep water. The convention is that the X-axis will be positive and increasing toward the shore while the positive Y-axis is 90 degrees to the left of the X-axis as shown in Figure 1. The grid interval is selected such that, in a given cell, the bottom contours are reasonably parallel to one another.

Actual or interpolated depths at the grid intersections are recorded to the best accuracy available from the chart, and all actual depths are made positive. Extrapolated depth values are continued on land for two grid units from the shoreline and are made negative. For depths on the shoreline itself, zero is used. Depths on land outside of two grid units from the shoreline are made some arbitrary, negative, non-zero value. The procedure for assigning negative values for the nearshore land depths is required in the fitting of a surface to the localized depth values. The zero land depths are used for contouring the shoreline on the output graph.

3. Example of Input.

As an illustration of the procedure and to test the results, the southern portion of Monterey Bay, California, was used. The depth grid was selected so that the origin of the 18 rays would be positioned in water depths greater than 1,024 feet where $d/L_0 = 0.5$ for a period of 20 seconds. The direction toward which the rays proceed is 125° TRUE, making

an angle with the X-axis (oriented at 90° TRUE) of - 35°. The grid interval selected was 1,500 feet, and the depth input was in fathoms to facilitate the chart reading. The depth values were determined from the U.S. Coast and Geodetic Survey chart 5403.

Other parameters that were required for input were the X and Y values of the starting point for the first ray. The values of the ray separation parameter, β , are input as B1 and B2, and are always equal to 1.0 in deep water. The period of the waves and the angle with respect to the X-axis are entered as T and A1, respectively. The following parameters are: NOR, the number of rays that are being followed; DIST, the distance between these rays; TIME, the time interval for advancement of the wave front; GRID, the grid interval; MM, the number of grid points in the X direction; and NN, the number of grid points in the Y direction. The angle of the wave direction is in relation to the X-axis and is the direction toward which the waves are moving. It can be positive or negative with respect to the X-axis.

4. Computer Operations.

The computer first reads the parameter values, then the depth grid as a column of Y-values for a constant X-value. The depths are converted to feet immediately. The subroutine DEPTFUN is called to compute the depth at the first point by fitting the closest nine grid point depths around the starting value to a quadric surface by the least squares method, using an equation of the form:

$$DEPTH = A_1 + A_2X + A_3Y + A_4XY + A_5X^2 + A_6Y^2$$

where DEPTH is the value of the depth at that point, $A_{1,2..n}$ are constant coefficients, and X and Y are the distance values for the point. Each

time a new depth is encountered the surface of best fit is computed from the surrounding nine grid points. Also included in the DEPTFUN subroutine is the computation of the wave velocity at that point for the given depth. This is done by an iteration process using the common result from solving the wave equation as used in H.O. Pub. 234 [2]:

$$C = \frac{gT}{2\pi} \tanh\left(\frac{2\pi d}{TC}\right)$$

where C = wave velocity, g = acceleration due to gravity, T = wave period, and d = water depth. As with other wave refraction investigations it is assumed that the wave velocity is a function only of water depth and wave period. Other factors such as bottom friction, percolation, currents, reflection, and wind are considered as not affecting the refracting waves.

The wave ray is moved to the next point by solving for the ray curvature and projecting the ray forward in the time interval specified at the speed calculated for the point.

The curvature is calculated using the expression [5]:

$$FK = \frac{1}{C}\left[\sin A \frac{\partial C}{\partial X} - \cos A \frac{\partial C}{\partial Y}\right]$$

where FK = ray curvature, and A = approach angle.

To determine the values of X_{n+1}, Y_{n+1}, A_{n+1}, and FK_{n+1} for the succeeding point, an iteration process is used to solve the equations [3]:

$$\Delta A = (FK_n + FK_{n+1})\, D/2$$

$$A_{n+1} = A_n + \Delta A$$

$$\bar{A} = (A_n + A_{n+1})/2$$

$$X_{n+1} = X_n + D \cos \bar{A}$$

$$Y_{n+1} = Y_n + D \sin \bar{A}$$

where D = the incremented distance (D = CT) between points n and n+1.

At the point n+1, the value of Beta is calculated by solving the second-order, non-linear differential equation [5]:

$$\frac{D^2\beta}{Ds^2} + p\frac{D\beta}{Ds} + q\beta = 0$$

where

$$p = -\cos A \frac{1}{c}\frac{\partial C}{\partial X} - \sin A \frac{1}{c}\frac{\partial C}{\partial Y}$$

$$q = \sin^2 A \frac{1}{c}\frac{\partial^2 C}{\partial X^2} - 2 \sin A \cos A \frac{1}{c}\frac{\partial^2 C}{\partial X \partial Y} + \cos^2 A \frac{1}{c}\frac{\partial^2 C}{\partial Y^2}$$

The above equation is solved by the finite difference method [3]. This results in an equation for the Beta value at the n+1 point in terms of the Beta values at the two previous points. The equation to be solved is then:

$$\beta_3 = \frac{(pD - 2)\beta_1 + (4 - 2qD^2)\beta_2}{2 + pD}$$

where p, q and D are defined above, and β_1 and β_2 are the Beta values of the two previous points. It is to be noted that the calculation of Beta is made for the n+1 point. In the BETA subroutine the coefficient of refraction is calculated by the relation:

$$K = \sqrt{1/\beta}$$

where K = the refraction coefficient.

To determine the ratio of wave height at the point to the deep-water wave height, the shoaling factor has been approximated for values

of $C/C_0 < 1$ by a curve such that [2]:

$$H_s = 3.2519 - 12.8150\left(\frac{C}{C_0}\right) + 28.81\text{[illegible]}2\left(\frac{C}{C_0}\right)^2 - 29.9257\left(\frac{C}{C_0}\right)^3 + 11.6815\left(\frac{C}{C_0}\right)^4$$

As C_0, the deep-water wave velocity, is a function of period alone and C is known for any point (X,Y), the equation may be evaluated at any or every point at which the calculations are made for C. The ratio of wave heights (H/H_0) is found from the equation [2]:

$$H/H_0 = H_s\sqrt{1/\beta}$$

5. Output from Computer.

The printed output from the computer consists of the X and Y values of each position after advancement by the increment of time specified. This is in units of yards for ease in hand plotting on the charts. The coefficient of refraction and the wave height ratio for each point are given.

Included in the program output is a graphical plot of the wave crests, which is programmed for the Calcomp 160 computer system, utilizing the DRAW subroutine of Appendix II. The first and every third crest thereafter are plotted and points connected depicting the wave crest. The scale is such that a true representation of the wave crest is presented. From this graph and if desired, hand plotting, areas of convergence and divergence are easily seen. By knowing the number of the crest, the parameters of refraction coefficient and wave height ratio can be found from the printed data.

6. Program Development.

To calculate the coefficient of refraction and the wave height ratio

it is necessary that the interpolation surface for the depth values be of the second order, at least, so that the second partial derivatives would be available for the computation of these values. Consequently, a quadric surface was used for the representation of the depth values at the closest nine grid points to the position at which the values were desired. The partial derivatives of the surface in the X and Y directions were used to evaluate the partial derivatives of the velocity function as proposed by Harrison and Wilson [4]. However, a power series representation of the hyperbolic tangent was used to evaluate the velocity derivatives rather than the method used by Harrison and Wilson, as shown in Appendix III.

The calculation of the Beta parameter is required for determining the refraction coefficient. The finite difference method is used in this program. However, the use of this method requires that the distances between the points at which the equation is being solved be equal. In the method presented here where the distance is a function of the velocity at each point, this does not hold exactly true. The difference in the velocities at the successive points is on the order of one foot per second due to the shallow contour gradient in the particular area of interest. This, of course, will not hold true for all cases. A better method would be to solve the second-order equation in Beta by the Kelvin method which requires only the distance between the point n and n+1, which is readily obtainable.

7. Discussion of Results.

The printed results are shown in Appendix I for several of the wave crests. The X and Y values from this type of presentation were plotted by hand as in Figure 1. This plot shows areas of marked divergence along the shore. When this figure is compared with Figure 2, the result of the

graphical method of H.O. Pub. 234, little difference is noted suggesting that both the computer method and the graphical method produce similar results. This is the aim of the investigation. Other comparisons were made using different directions and periods with comparable results.

In Figure 1, the seventh and eighth wave rays are seen to cross and continue to the shore. This crossing is attributed to the bottom contours of ray eight when the ray first reaches shallow water ($d/L_0 = 0.5$). There is a small but steep-sided indentation in the otherwise gradual slope of the area. This causes the ray to bend due to the steep depth gradient which results in convergence with ray seven below it. From this point to the shore the refraction is similar to that of ray six. Figure 3, the computer drawn plot, shows all of the wave crests from the first advancement to the shore. The ray crossing is evident.

The values of the coefficient of refraction and the wave height ratio, Appendix I, show little refraction for crests one and two as expected, since the waves for the most part are in deep water. However, later values of these parameters do show the effects of the refraction seen in Figures 1 and 3. The values of these parameters require verification. The values do appear to be qualitatively reasonable for the refraction encountered, and when compared to the values estimated by the technique of measuring the distance between orthogonals.

It is estimated that the time necessary to construct the grid and record the depth values is three to four hours depending on the size of the area desired and the gradient of the bottom contours in that area. A shallower gradient requires a larger area to ensure that the wave rays start in deep water. The time required to punch the data cards and to check the results is a function of the experience of the operator. The

computer time is approximately five minutes for the computation of the results in this paper, Appendix I, plus an additional five minutes to draw the graph. This time can be reduced drastically if the program as compiled by the computer is put on tape and input in that form. The compiler takes over half of the computer time at present. The time is spent constructing the grid of depth values only once. From this grid all of the various wave periods and directions can be investigated without further effort of constructing another grid as is necessary in the hand plotting of the refraction diagram.

8. Conclusion.

As discussed in the preceeding section, the results of the refraction diagram agree well with that of the hand drawn method, so that the plots received from the computer can be used with as much confidence as those drawn manually. The decided advantage is that the computer product can be obtained many times faster.

The coefficient of refraction and the wave height ratio as noted appear qualitatively correct and can be used with the reservation that the values have not yet been verified.

9. Recommendations.

Further development of programs similar to this will require a better representation of the bottom contours for more accurate results. The shallow-water depth values need to be very accurate for the proper determination of all the parameters, since the depth determines the velocity of the wave at each point, and the other parameters are functions of the velocity.

An accurate method within the approximations made is required for the calculation of the Beta parameter. The analytic approach of Kelvin's

method appears to be the more accurate method from the consideration of the necessary assumptions involved with the finite difference method. In the case involving steep gradients of the bottom contours, the finite difference method as used here may prove inadequate. The time factor in completing this paper prevented the successful completion of programming the Kelvin method as desired.

Testing of the results is required to substantiate the procedure as a prediction method. Involved aerial photography is required to follow the wave crest refraction to the shoreline where the accurate prediction is most desired. Wave gages and other devices may be used to obtain the necessary data for the verification of the wave height values as predicted by the computer. A very intensive project will be required in order to verify the predicted values.

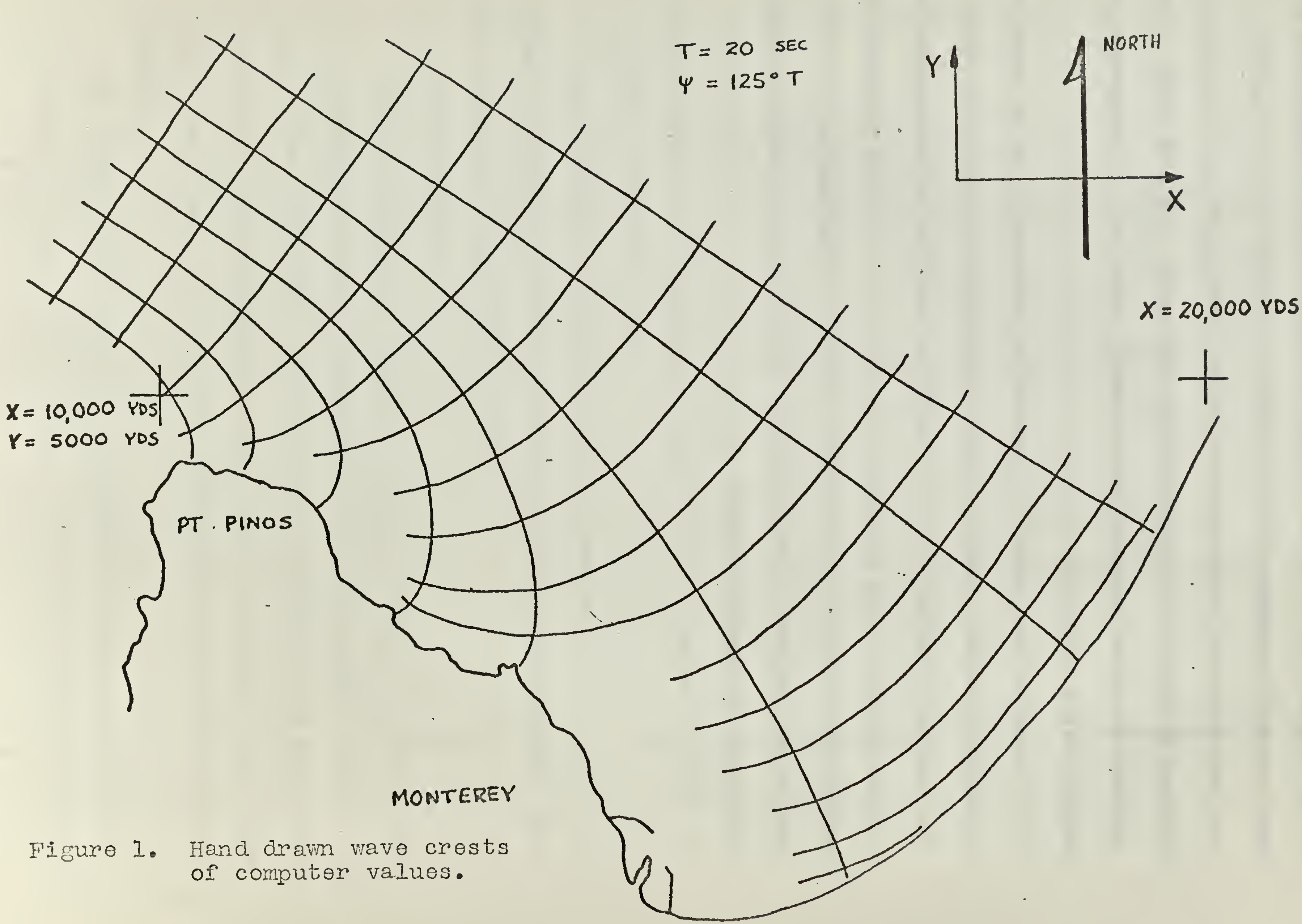

Figure 1. Hand drawn wave crests of computer values.

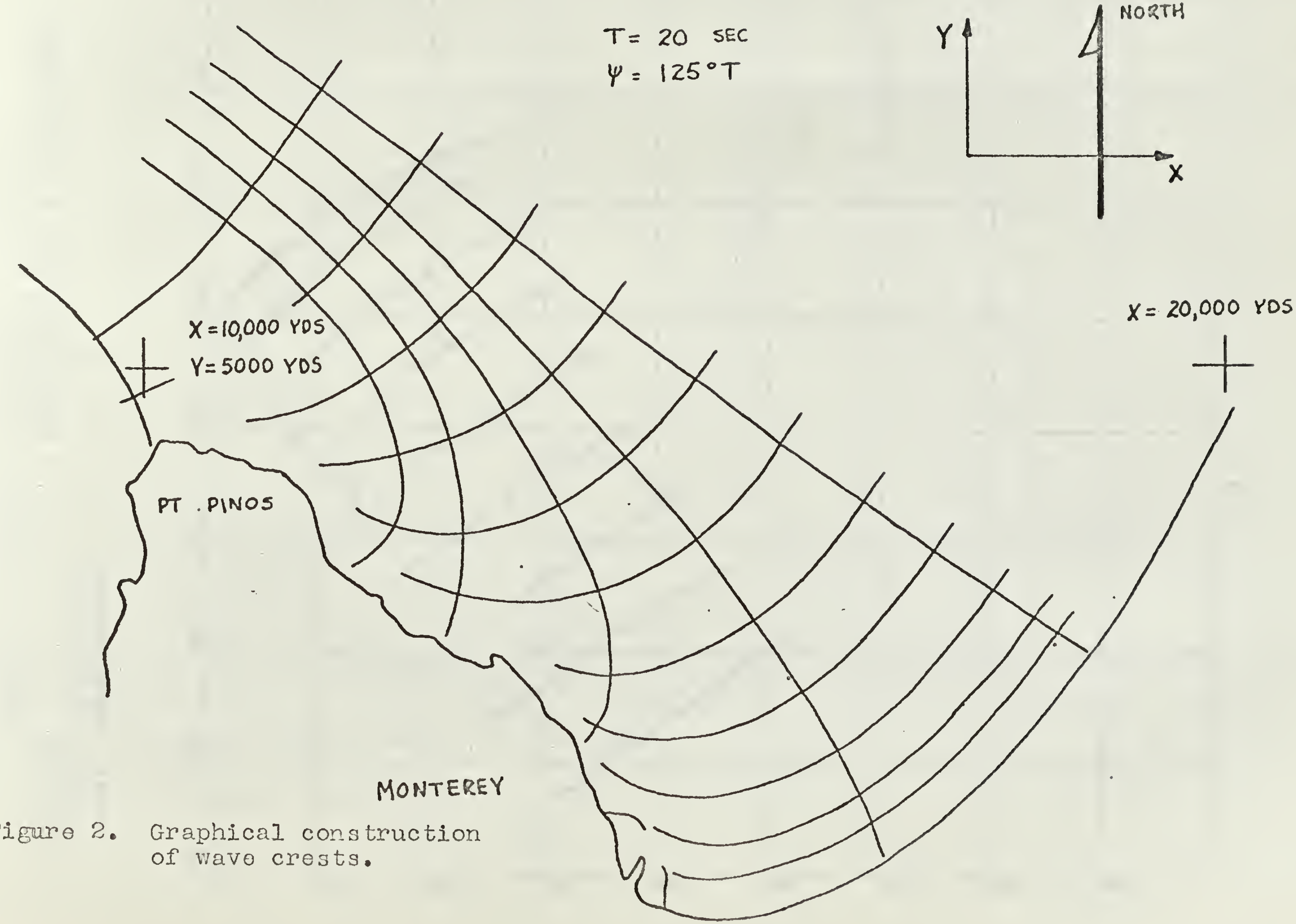

Figure 2. Graphical construction of wave crests.

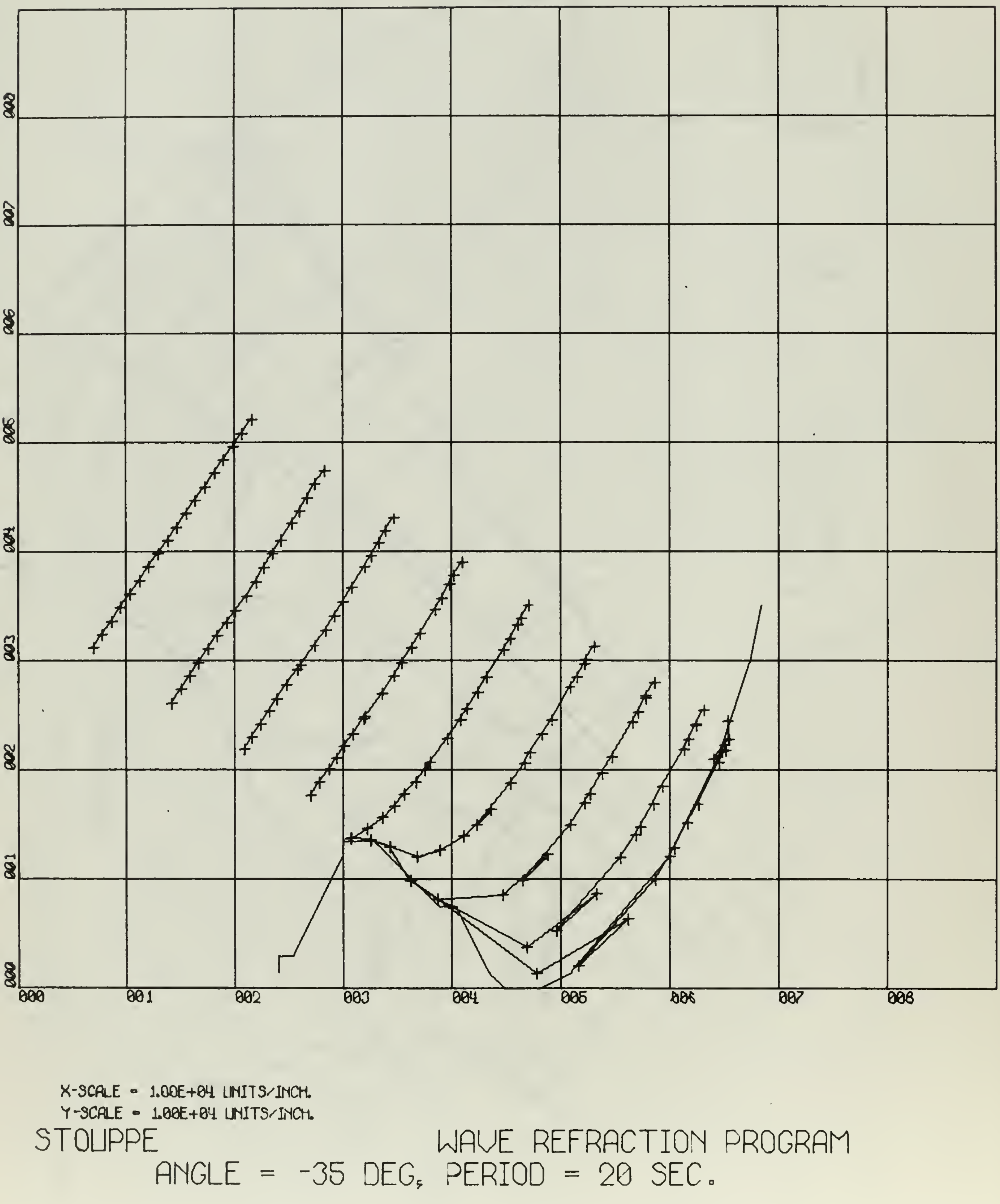

Figure 3. Computer drawn plot of wave crests

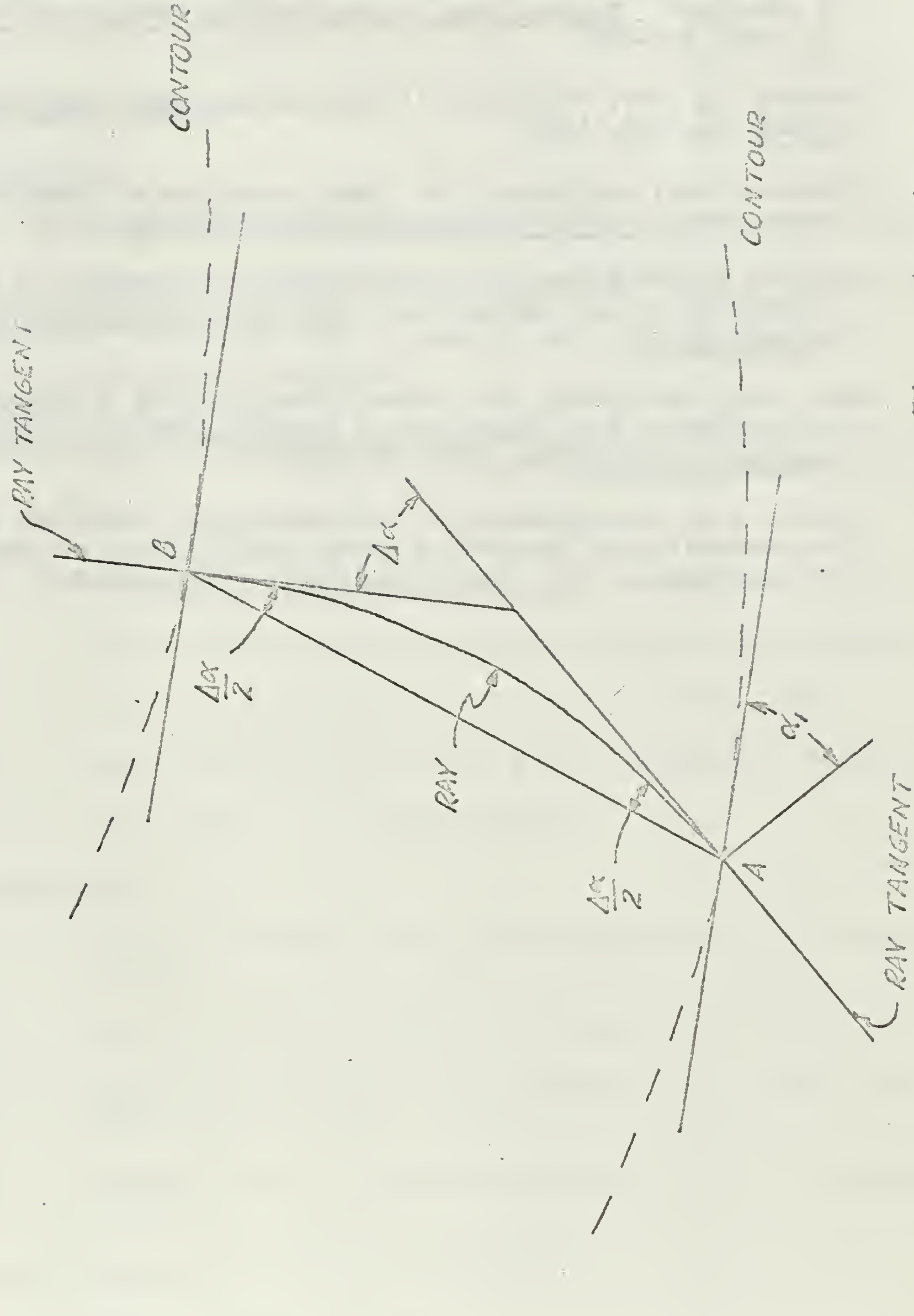

Figure 4. The ray tangents as depth contours are crossed.

BIBLIOGRAPHY

1. Arthur, R.S., Munk, W.H., and Isaacs, J.D. The direct construction of wave rays. Transactions of the American Geophysical Union, v. 33, 1952: 855-865.

2. Breakers and surf, principles in forecasting. U.S. Navy Hydrographic Office Pub. 234, 1944.

3. Griswold, G.M., and Nagle, F.W. Wave refraction by numerical methods. Mimeo. Rept., U.S. Navy Weather Research Facility, 1962.

4. Harrison, W. and Wilson, W.S. Development of a method for numerical calculation of wave refraction. U.S. Army Coastal Engineering Research Center, T.M. 6, 1964.

5. Munk, W.H., and Arthur, R.S. Wave intensity along a refracted ray. Gravity Waves, U.S. Department of Commerce, National Bureau of Standards Circular 521, 1952: 95-108.

6. Pierson, W.J. Jr., Newmann, G., and James, R.W. Practical methods for observing and forecasting ocean waves by means of wave spectra and statistics. U.S. Navy Hydrographic Office Pub 603, 1955.

APPENDIX I

Computer Program for Wave Refraction using Fortran 63

on the C.D.C. 1604 Computer

Program Title: WAVREFR

Input Variables:

XV, YV ..... X and Y values of the first wave ray to be computed (feet).

B1, B2...... Values of the Beta coefficient (always equal to one in deep water).

T .... Wave period (seconds).

A1 .... Angle measured from the direction of increasing X along the X-axis in the direction of travel of the wave crest (degrees).

NOR .... Number of rays that will be followed.

DIST .... Perpendicular distance between rays in deep water (feet).

TIME .... Time interval used for advancing the wave front (seconds).

GRID .... Distance between grid points of depth values (feet).

MM, NN ..... Number of grid points in the X and Y directions, respectively.

DEP (I,J)... Depth values in the grid (fathoms).

Output Variables:

X, Y .... X and Y coordinates for a given wave crest and ray number (yards).

COREFR ..... Coefficient of refraction for the wave at the point X,Y.

HHO .... Ratio of wave height to the deep-water wave height at the point X,Y.

NGO .... Indicates that the ray has terminated (= 1), or is continuing (= 2).

Variables in Common: (not previously defined)

FK, FFK .... Values of ray curvature.

CXY, CXX ... Values of wave velocity.

PAR .... Constant = $gT/2\pi$.

BAR .... Constant = $2\pi/T$.

MAX .... Number of wave crest (= 1 for first crest).

DEPTH ...... Calculated depth at point X, Y.

NORA .... Number of ray being followed.

LO .... Indicates ratio of d/L_0.

= 1 for $d/L_0 > 0.5$;

= 2 for $d/L_0 \leq 0.5$.

MIT .... Designates whether the last two curvature estimates for a given X, Y are less than .0001, whether the 13th and 15th estimates are less than .0001, or whether neither of the above is true (MIT = 1, 2, 3, for the respective cases).

NIP .... Used to determine the number of wave crest to be graphed.

XXX, FX .... Values of X used in the graphing and printout.

YYY, FY .... Similar to uses for XXX and FX.

NNGO .... The number of rays that have stopped.

XX, YY, AA, A

..... Intermediate values of X, Y, and A.

Summary of Program:

The program reads the input variables, then the depth grid. PAR and BAR are computed for the wave period; A is converted to radians; and MAX and NORA are set equal to one. An initial value of CXY is found by testing the wave period. Control is transferred to the RAYN subroutine. When RAYN has determined the parameters at the first and second points along the ray, NORA is increased one, the XX and YY values computed for the next ray, and then RAYN is called again to compute the parameters. This procedure is followed until NORA equals NOR, the number of rays. The graph is drawn for the first wave crest and the values printed. MAX is increased by one, and the procedure started over until all the rays have

stopped by either going off the grid, hitting the shoreline, or not having the values of the curvature converge. When all the rays have stopped NNGO is equal to NOR. The final graph is drawn which is the contour of the area utilizing the zero values of the depth grid and the program is complete.

Subroutine Title: RAYN

Variables of Subroutine:

COREFA ..... Intermediate value of COREFR.

HHA .... Intermediate value of HHO.

FKBAR ...... Curvature used to obtain DEL A.

NOGO .... Storage value of NGO.

AAA .... Storage value of A.

Summary of Subroutine:

RAYN calls DEPTFUN to obtain CXY, PDPX, PDPY, PDDPXY, PDDPXX, and PDDPYY. NGO is set equal to two if DEPTH is not zero, otherwise NGO is set equal to one. RAYN tests NGO of the ray for the MAX calculation to determine whether the ray will continue or not. If NGO = 2, KFUNCT is called to obtain the value of the curvature, FK. If NGO = 1, control is returned to WAVREFR. If the ray is continuing RAYN calls MOVE to project the ray to its next position. If the ray is not stopped in the MOVE subroutine, RAYN next calls BETA to calculate the values of COREFR and HHO. As a final step RAYN stores the values of FK, COREFR, HHO, NGO, X, Y, PDPC, and AAA for use when the ray is again projected.

Subroutine Title: DEPTFUN

Variables of Subroutine:

KER .... Indicates errors in the solution of the simultaneous equations for the quadric surface:

= 1 no errors indicated in the solution;

= 2 indicates that the matrix of values being solved is singular or nearly singular.

ALO .... Deep-water wave length.

DLO .... Ratio of DEPTH to ALO.

PDPX, PDPY, PDDPXX, PDDPXY, PDDPYY

..... First and second partial derivatives of the depth at X, Y with respect to the X and Y directions.

Summary of Subroutine:

The subroutine first determines the values of the closest grid point by truncating the values of XX/GRID + 1.5 and YY/GRID + 1.5 to give the correct value of the grid point. The quadric surface is fitted to the nine values of depth surrounding this calculated point, using the least squares method. DEPTH is found by evaluating the quadric equation at X and Y values of the point. If DEPTH is positive, the wave velocity, CXY, is solved for by an iteration procedure of the equation described in the test using the principles of Appendix IV. The various partial derivatives are computed by evaluating the equation described in Appendix III. The method for the solution and the program for solving the six simultaneous equations resulting from the least squares method was written by C. B. Bailey and Mary Haynes of the USNPGS computer center programming staff.

Subroutine Title: KFUNCT

Variables of Subroutine:

PDPC, PDDPCC

.... First and second partial derivatives of the depth with respect to the wave velocity.

PCPX, PCPY .. First derivatives of the wave velocity with respect to the X and Y directions calculated from relations in Appendix III.

FK .... Curvature of the ray at the point X,Y.

Summary of Subroutine:

The subroutine calculates the values of PCPX and PCPY, and the curvature at the point X, Y from the equations of Appendix III and the text, respectively. The curvature is calculated only if LO = 2, where the wave is in shallow water.

Subroutine Title: MOVE

Variables of Subroutine:

FFKK .... Storage value of FKBAR.

DEL D ...... Increment of distance advanced (DEL D = T CXY).

DEL X ....XX - X.

DEL Y ...... YY - Y.

DEL A ...... AA - A.

ABAR .... (A + AA)/2.

Summary of Subroutine:

MOVE determines the X and Y values of the next point on the wave ray by an iteration process involving the curvature, the incremented distance, and the angle of the ray. The iteration is continued until the curvature estimates vary no more than .0001 from one another. Then the values of XX, YY, AA, and FKK are accepted for the new point. If the difference is greater, FKKP is set equal to FKBAR and the current FKBAR is used to obtain another set of values. If the iteration process stops before 15 iterations, MIT = 1. If the cycle stops at 15 iterations, and the difference between FKBAR and FKKP is less than .0001, MIT = 2, and FKBAR is defined as (FKBAR + FKKP)/2 for the last determination of XX, YY, AA, and FKK values. If the difference is greater than .0001 after 15 iterations, MIT = 3, and the ray is stopped. Control is sent back to RAYN. Inside the iteration loop DEPTFUN determines the parameters of CXY, PDPX, and the other derivatives, and tests the values of X and Y to determine if they are still on the grid.

Subroutine Title: BETA

Variables of Subroutine:

PCCPXX, PCCPXY, PCCPYY

.... Second partial derivatives of the wave velocity with respect to the X and Y directions.

HSHOL ...... Shoaling factor.

CCO .... Ratio of the wave velocity to the deep-water wave velocity.

DD .... Increment of distance along the wave ray between points n and n+1.

B(i,j) ..... Values of the Beta parameter at the three points.

Summary of Subroutine:

BETA calculates the values of the coefficient of refraction and the wave height ratio at each point along the ray. The equations are described in the text. If LO = 1, BETA sets COREFR and HHO equal to one, and the Beta value equal to the previous Beta value on the ray. The shoaling factor is calculated from an equation determined from a polynomial fit to a curve of $C/C_0 < 1$ as described in the text.

```
      PROGRAM WAVREFR                                                   WAVE0000
C     WAVE REFRACTION PROGRAM FOR THE COMPUTATION OF CREST AND RAY      WAVE0010
C     REFRACTION IN SHAOLING WATER BY LT. D. STOUPPE, OCTOBER 1966.     WAVE0020
C     THE PROGRAM IS WRITTEN IN FORTRAN 63 FOR THE CDC 1604 COMPUTER    WAVE0030
C     SYSTEM. INPUT OF THE BINARY IS REQUIRED FOR THE DRAW SUBROUTINE   WAVE0040
C     ON THE PRESENT SYSTEM AT THE US NAVAL POSTGRADUATE SCHOOL.        WAVE0050
C     NOTE TO USERS  ALL INTEGER INPUT ON THE DATA CARDS MUST BE        WAVE0060
C     RIGHT ADJUSTED FOR CORRECT  RESULTS.                              WAVE0070
C     USERS MAY ENTER THE DESIRED NAMES FOR THE GRAPH TITLES IN THE     WAVE0080
C     ITITLE STATEMENTS AS DESCRIBED IN THE WRITEUP ON THE DRAW         WAVE0090
C     SUBROUTINE.                                                       WAVE0100
      DIMENSION ITITLE(12),XXX(100),YYY(100),FX(100),FY(100)            WAVE0110
      COMMON/BLK1/X(100),Y(100),COREFR(100),HHO(100),B(3,100),FFK(100), WAVE0120
     1NOGO(100),NGO,AAA(100)                                            WAVE0130
      COMMON/BLK2/T,A,CXY,PAR,BAR,TIME,GRID,FK,MAX,NOR,MM,NN,DEPTH,HHA, WAVE0140
     1COREFA,NORA,LO,DIST                                               WAVE0150
      COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY        WAVE0160
     1,PDDPXY,PDDPCC,PDPC                                               WAVE0170
      READ2,XV,YV,B1,B2,T,A1,NOR,DIST,TIME,GRID,MM,NN                   WAVE0180
    2 FORMAT (2F10.1,2F3.0,F4.0,F4.0,I4,F10.2,F5.1,F10.1,2I4)           WAVE0190
      PRINT 19                                                          WAVE0200
   19 FORMAT(1H1,5X,2HXV,7X,2HYV,4X,2HB1,1X,2HB2,3X,1HT,2X,2HA1,2X,     WAVE0210
     13HNOR,3X,4HDIST,4X,4HTIME,3X,4HGRID,3X,2HMM,2X,2HNN/)             WAVE0220
      PRINT 2,XV,YV,B1,B2,T,A1,NOR,DIST,TIME,GRID,MM,NN                 WAVE0230
      READ 1,((DEP(I,J),J=1,NN),I=1,MM)                                 WAVE0240
    1 FORMAT (14F5.0)                                                   WAVE0250
      DO22 I=1,MM $ DO22 J=1,NN                                         WAVE0260
   22 DEP(I,J)=DEP(I,J)*6.                                              WAVE0270
      MAX=1 $ A1=A1*.01745329 $ PAR=32.2*T/6.283185 $ BAR=6.283185/T    WAVE0280
      NIP=4                                                             WAVE0290
      LABEL=4H         $ DO 18 M=1,12                                   WAVE0300
   18 ITITLE(M)=8H           $ ITITLE(1)=8H STOUPPE $ ITITLE(4)=8HWAVE REFWAVE0310
      ITITLE(5)=8HRACTION    $ ITITLE(6)=8HPROGRAM                      WAVE0320
      ITITLE(8)=8HANGLE =   $ ITITLE(10)=8H PERIOD                      WAVE0330
      ITITLE(9)=8H-35 DEG,    $ ITITLE(11)=8H= 20 SEC                   WAVE0340
      B(1,1)=B1 $ B(2,1)=B2 $ XX=XV $ YY=YV $ NORA=1 $ A=A1             WAVE0350
      IF(T-10.)15,16,17                                                 WAVE0360
   15 CXY= 30.0 $ GO TO 3                                               WAVE0370
   16 CXY= 50.0 $ GO TO 3                                               WAVE0380
   17 CXY=80.0 $ GO TO 3                                                WAVE0390
    5 B(1,NORA)=B1 $ B(2,NORA)=B2 $ A=A1                                WAVE0400
      XV=XV-DIST*SINF(A) $ YV= YV+DIST*COSF(A)$XX=XV $ YY=YV $ GO TO 3  WAVE0410
   23 NORA=1                                                            WAVE0420
   24 XX=X(NORA) $ YY=Y(NORA) $ A=AAA(NORA)                             WAVE0430
    3 CALL RAYN(XX,YY,NOR)                                              WAVE0440
      CONTINUE $ NORA=NORA+1 $ IF(NORA-NOR)10,10,4                      WAVE0450
   10 IF(MAX-1)5,5,24                                                   WAVE0460
    4 IF(MAX-1)20,20,25                                                 WAVE0470
   25 IF(NIP-4)21,26,26                                                 WAVE0480
   20 CALL DRAW(NOR,X,Y,1,2,LABEL,ITITLE,10000.,10000.,0,0,2,2,9,9 ,1,  WAVE0490
     1LAST) $ GO TO 27                                                  WAVE0500
   26 CALL DRAW(NOR,X,Y,2,2,LABEL,ITITLE,10000.,10000.,0,0,2,2,9,9 ,1,  WAVE0510
     1LAST)                                                             WAVE0520
```

```
27 CALL DRAW(NOR,X,Y,2,0,LABEL,ITITLE,10000.,10000.,0,0,2,2,9,9 ,1,  WAVE0530
  1 LAST)                                                            WAVE0540
   NIP=1                                                             WAVE0550
21 NIP=NIP+1 $ MAX=MAX+1 $ MIN=MAX-1                                 WAVE0560
33 DO 32 M=1,NOR $ FX(M)=X(M)/3. $ FY(M)=Y(M)/3.                     WAVE0570
32 CONTINUE                                                          WAVE0580
34 PRINT 28,MIN                                                      WAVE0590
28 FORMAT(//17HNUMBER OF CREST = I4/)                                WAVE0600
   PRINT 6                                                           WAVE0610
 6 FORMAT(/4X,1HX,12X,1HY,8X,6HCOREFR,3X,3HHHO,7X,3HNGO,10X,1HX,12X, WAVE0620
  11HY,8X,6HCOREFR,3X,3HHHO,7X,3HNGO/)                               WAVE0630
   DO 7 J=1,NOR,2                                                    WAVE0640
   PRINT 8,FX(J),FY(J),COREFR(J),HHO(J),NOGO(J),FX(J+1),FY(J+1),     WAVE0650
  1COREFR(J+1),HHO(J+1),NOGO(J+1)                                    WAVE0660
 8 FORMAT(2(F10.2,3X,F10.2,3X,F5.2,2X,F5.2,3X,I5,7X)/)               WAVE0670
 7 CONTINUE                                                          WAVE0680
   DO 9 I=1,NOR $ DO 9 J=2,3                                         WAVE0690
 9 B(J-1,I)=B(J,I) $ NNGO=0 $ DO11 K=1,NOR                           WAVE0700
   IF(NOGO(K)-1)12,12,11                                             WAVE0710
12 NNGO=NNGO+1                                                       WAVE0720
11 CONTINUE $ IF(NNGO-NOR) 13,14,14                                  WAVE0730
13 GO TO 23                                                          WAVE0740
14 CONTINUE                                                          WAVE0750
   M=1 $ DO31I=1,MM $ DO31J=1,NN $ IF(DEP(I,J))31,30,31              WAVE0760
30 XXX(M)=(I-1)*GRID $ YYY(M)=(J-1)*GRID $ M=M+1 $ N=M-1             WAVE0770
31 CONTINUE                                                          WAVE0780
   CALL DRAW(N,XXX,YYY,3,0,LABEL,ITITLE,10000.,10000.,0,0,2,2,9,9 ,  WAVE0790
  11,LAST)                                                           WAVE0800
   END                                                               WAVE0810
   SUBROUTINE RAYN(XX,YY,NOR)                                        RAYN0820
   COMMON/BLK1/X(100),Y(100),COREFR(100),HHO(100),B(3,100),FFK(100), RAYN0830
  1NOGO(100),NGO,AAA(100)                                            RAYN0840
   COMMON/BLK2/T,A,CXY,PAR,BAR,TIME,GRID,FK,MAX,ABC,MM,NN,DEPTH,HHA, RAYN0850
  1COREFA,NORA,LO,DIST                                               RAYN0860
   COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY        RAYN0870
  1,PDDPXY,PDDPCC,PDPC                                               RAYN0880
   COMMON/BLK4/FFKK(100),DLC,C(6),MIT,DEL X,DEL Y                    RAYN0890
   IF(MAX-1)110,110,111                                              RAYN0900
110 NGO=2 $ GO TO 108                                                RAYN0910
111 NGO=NOGO(NORA) $ GO TO (104,108)NGO                              RAYN0920
108 CALL DEPTFUN(XX,YY)                                              RAYN0930
   GO TO (104,102)NGO                                                RAYN0940
104 NGO=1$COREFA=0.0 $B(3  ,NORA)=0.0$FK=0.0 $ HHA=0.0 $ GO TO 103   RAYN0950
102 IF(MAX-1)105,105,107                                             RAYN0960
105 CALL KFUNCT(A,FK) $ GO TO 106                                    RAYN0970
107 FK=FFK(NORA)                                                     RAYN0980
106 CALL MOVE(XX,YY) $ GO TO(104,109)NGO                             RAYN0990
109 CALL BETA(XX,YY)                                                 RAYN1000
103 FFK(NORA)=FK $ COREFR(NORA)=COREFA $ HHO(NORA)                   RAYN1010
  1=HHA $ NOGO(NORA)=NGO $ X(NORA)=XX $ Y(NORA)=YY $ AAA(NORA)=A     RAYN1020
   RETURN                                                            RAYN1030
   END                                                               RAYN1040
   SUBROUTINE DEPTFUN(XX,YY)                                         DEPT1050
```

```
      DIMENSION D(6,6),E(6),      O(100),P(100)
      COMMON/BLK1/X(100),Y(100),COREFA(100),HHO(100),R(3,100)FFK(100),
     1NOGO(100),NGO,AAA(100)
      COMMON/BLK2/T,A,CXY,PAR,BAR,TIME,GRID,FK,MAX,NOR,MM,NN,DEPTH,HHA,
     1COREFA,NORA,LO,DIST
      COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY
     1,PDDPXY,PDDPCC,PDPC
      COMMON/BLK4/FFKK(100),DLO,C(6),MIT,DEL X,DEL Y
      XR=XX/GRID $ K=XR+1.5 $ YR=YY/GRID $ L=YR+1.5
      IF((XX-1.0)*((MM-1)-K))62,7,7
    7 IF((YY-1.0)*((NN-1)-L))62,6,6
    6 DO9J=1,6 $ DO9I=1,6
    9 D(I,J)=0.0 $ DO8 I=1,6
    8 E(I)=0.0
      M=L-1 $ MA=L+1 $ N=K-1 $ NA=K+1
      DO10 J=M,MA $ DO10 I=N,NA $ O(I)=(I-1)*GRID $ P(J)=(J-1)*GRID
      D(1,1)=9.$D(2,1)=D(2,1)+O(I)$D(3,1)=D(3,1)+P(J)$D(4,1)=D(4,1)+O(I
     1)*P(J)$D(5,1)=D(5,1)+O(I)**2$D(6,1)=D(6,1)+P(J)**2$E(1)=E(1)+DEP
     2(I,J)$D(1,2)=D(2,1)$D(2,2)=D(5,1)$D(3,2)=D(4,1)$D(4,2)=D(4,2)+O(I
     3)**2*P(J)$D(5,2)=D(5,2)+O(I)**3$D(6,2)=D(6,2)+O(I)*P(J)**2$E(2)=
     4E(2)+O(I)*DEP(I,J)$D(1,3)=D(3,1)$D(2,3)=D(4,1)$D(3,3)=D(6,1)$D(4,
     53)=D(4,3)+O(I)*P(J)**2$D(5,3)=D(4,2)$D(6,3)=D(6,3)+P(J)**3$E(3)=
     6E(3)+P(J)*DEP(I,J)$D(1,4)=D(4,1)$D(2,4)=D(4,2)$D(3,4)=D(6,2)$D(4,
     74)=D(4,4)+O(I)**2*P(J)**2$D(5,4)=D(5,4)+O(I)**3*P(J)$D(6,4)=D(6,4
     8)+O(I)*P(J)**3$E(4)=E(4)+O(I)*P(J)*DEP(I,J)$D(1,5)=D(5,1)$D(2,5)=
     9D(5,2)$D(3,5)=D(4,2)$D(4,5)=D(5,4)$D(5,5)=D(5,5)+O(I)**4$D(6,5)=
     1D(4,4)$E(5)=E(5)+O(I)**2*DEP(I,J)$D(1,6)=D(6,1)$D(2,6)=D(6,2)$D(3
     2,6)=D(6,3)$D(4,6)=D(6,4)$D(5,6)=D(4,4)$D(6,6)=D(6,6)+P(J)**4$E(6)
     3=E(6)+P(J)**2*DEP(I,J)
   10 CONTINUE
      NPM=7  $ DO 34M=1,6 $ KP=0 $ Z=0.0 $ DO12N=M,6 $ IF(Z-ABSF(D(N,M)
     1))11,12,12
   11 Z=ABSF(D(N,M)) $ KP=N
   12 CONTINUE $ IF(M-KP)13,20,20
   13 DO14J=M,NPM $Z=D(M,J) $ D(M,J)=D(KP,J)
   14 D(KP,J)=Z
   20 IF(ABSF(D(M,M))-.00001)50,50,30
   30 IF(M-6)31,40,40
   31 LP1=M+1 $ DO34N=LP1,6 $ IF(D(N,M))32,34,32
   32 RATIO=D(N,M)/D(M,M)$ DO33J=LP1,NPM
   33 D(N,J)=D(N,J)-RATIO*D(M,J)
   34 CONTINUE
   40 DO43I=1,6 $ II=7-I $           JPN=7   $ S=0.0 $ IF(II-6)41,43,
     143
   41 IIP1=II+1 $ DO 42N=IIP1,6
   42 S=S+D(II,N)*C(N)
   43 C(II)  =(D(II,JPN)-S)/D(II,II) $ KER=1 $ GO TO 51
   50 KER=2
      PRINT 54,XX,YY,KER
   54 FORMAT(/3X,5HXX = 1PE20.8,3X,5HYY = 1PE20.8,4X,6HKER = I2,
     117H  MATRIX SINGULAR/)
   51 GO TO (52,53)KER
   53 RETURN
```

```
52 CONTINUE
   DEPTH=C(1)+C(2)*XX+C(3)*YY+C(4)*XX*YY+C(5)*XX**2+C(6)*YY**2
   IF(DEPTH)62,62,65
65 ALO=PAR*T $ DLO=DEPTH/ALO $ IF(DLO-0.5)63,63,66
66 CXY=PAR $ LO=1 $ NGO=2 $ FK=0. $ GO TO 67
63 DO60M=1,50 $ CXX=PAR*TANHF(BAR*DEPTH/CXY)
   IF(ABSF(CXX-CXY)-0.01)61,61,60
60 CXY=(CXX+CXY)*.5
61 CONTINUE  $ NGO=2 $ LO=2
67 PDPX=C(2)+C(4)*YY+2*C(5)*XX
   PDPY=C(3)+C(4)*XX+2*C(6)*YY
   PDDPXY=C(4) $ PDDPXX=2*C(5) $ PDDPYY=2*C(6)
   GO TO 64
62 NGO=1 $ GO TO 81
64 CONTINUE
81 RETURN
   END
   SUBROUTINE KFUNCT (A,FK)
   COMMON/BLK1/X(100),Y(100),COREFR(100),HHO(100),B(3,100),FFK(100),
  1NOGO(100),NGO,AAA(100)
   COMMON/BLK2/T,UU,CXY,PAR,BAR,TIME,GRID,VV,MAX,NOR,MM,NN,DEPTH,HHA
  1,COREFA,NORA,LO,DIST
   COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY
  1,PDDPXY,PDDPCC,PDPC
 3 GO TO (5,6)LO
 6 R1=CXY/32.2 $ R2=R1**3*BAR**2 $ R3=R1**5*BAR**4 $ R4=R1**7*BAR**6
   PDPC=2.*R1+4.*R2/3.+6.*R3/5.+8.*R4/7.
   PDDPCC=(2.*R1+4.*R2+6.*R3+8.*R4)/CXY
   PCPX=PDPX/PDPC $ PCPY=PDPY/PDPC
   FK=(PCPX*SINF(A)-PCPY*COSF(A))/CXY $ GO TO 4
 5 FK=0.
 4 RETURN
   END
   SUBROUTINE MOVE(X,Y)
   COMMON/BLK1/U(100),V(100),COREFR(100),HHO(100),B(3,100),FFK(100),
  1NOGO(100),NGO,AAA(100)
   COMMON/BLK2/T,A,CXY,PAR,BAR,TIME,GRID,FK,MAX,NOR,MM,NN,DEPTH,HHA,
  1COREFA,NORA,LO,DIST
   COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY
  1,PDDPXY,PDDPCC,PDPC
   COMMON/BLK4/FFKK(100),DLO,C(6),MIT,DEL X,DEL Y
   FKBAR=FFKK(NORA)
   IF(MAX-1)1,1,4
 1 FKBAR=FK
 4 MIT=1
   DEL D=TIME*CXY
   GO TO (22,21)LO
22 XX=X+DELD*COSF(A) $ YY=Y+DELD*SINF(A) $ AA=A $ FKK=0. $ FKBAR=0.
   GO TO 6
21 DO 20IT=1,15
19 DEL A=FKBAR*DEL D $ AA=A+DEL A $ ABAR=A+.5*DEL A $ DEL X=DEL D*
  1 COSF(ABAR) $ DEL Y=DEL D*SINF(ABAR) $ XX=X+DEL X $ YY=Y+DEL Y
   GO TO (101,6)MIT
```

```
101 CALL DEPTFUN(XX,YY)      $ GO TO(38,10)NGO                       MOVE2120
 10 CALL KFUNCT(AA,FKK) $ FKBAR=.5*(FK+FKK)                          MOVE2130
    IF(IT-13)5,37,9                                                  MOVE2140
 37 FKKPP=FKBAR                                                      MOVE2150
  5 IF(MAX-1)7,7,9                                                   MOVE2160
  7 IF(IT-1)20,20,9                                                  MOVE2170
  9 IF(ABSF(FKKP-FKBAR)-.00001)6,6,20                                MOVE2180
 20 FKKP=FKBAR                                                       MOVE2190
 24 IF(ABSF(FKKPP-FKBAR)-.00001)18,18,17                             MOVE2200
 17 MIT=3 $ NGO=1 $ GO TO38                                          MOVE2210
 18 FKBAR=.5*(FKBAR+FKKP) $ MIT=2 $ GO TO 19                         MOVE2220
  6 NGO=2 $ GO TO 8                                                  MOVE2230
  8 X=XX $ Y=YY $ A=AA $ FK=FKK                                      MOVE2240
 38 CONTINUE                                                         MOVE2250
    FFKK(NORA)=FKBAR                                                 MOVE2260
    RETURN                                                           MOVE2270
    END                                                              MOVE2280
    SUBROUTINE BETA(XX,YY)                                           BETA2290
    COMMON/BLK1/X(100),Y(100),COREFR(100),HHO(100),B(3,100),FFK(100),BETA2300
   1NOGO(100),NGO,AAA(100)                                           BETA2310
    COMMON/BLK2/T,A,CXY,PAR,BAR,TIME,GRID,FK,MAX,NOR,MM,NN,DEPTH,HHA,BETA2320
   1COREFA,NORA,LO,DIST                                              BETA2330
    COMMON/BLK3/DEP(100,100),PDPX,PDPY,PCPX,PCPY,PDDPXX,PDDPYY       BETA2340
   1,PDDPXY,PDDPCC,PDPC                                              BETA2350
    COMMON/BLK4/FFKK(100),DLO,C(6),MIT,DEL X,DEL Y                   BETA2360
    GO TO (5,6)LO                                                    BETA2370
  5 COREFA=1. $ HHA=1. $ B(3,NORA)=B(2,NORA) $ COREFB=1. $ HHB=1.    BETA2380
    GO TO 7                                                          BETA2390
  6 PCCPXX=PDDPXX/(PDPC+PDDPCC)                                      BETA2400
    PCCPYY=PDDPYY/(PDPC+PDDPCC)                                      BETA2410
    PCCPXY=PDDPXY/(PDPC+PDDPCC)                                      BETA2420
    P=(-COSF(A)*PCPX-SINF(A)*PCPY)/CXY  $ Q=(SINF(A)**2*PCCPXX-2.*   BETA2430
   1SINF(A)*COSF(A)*PCCPXY+COSF(A)**2*PCCPYY)/CXY                    BETA2440
    DD=SQRTF((DEL X)**2+(DEL Y)**2)                                  BETA2450
    B(3,NORA)=(B(1,NORA)*(P*DD-2.)+B(2,NORA)*(4.-2.*DD**2*Q))        BETA2460
   1/(2.+P*DD)                                                       BETA2470
    COREFA=1./SQRTF(ABSF(B(2,NORA)))                                 BETA2480
    CCO=CXY/PAR $ HSHOL=3.2519-12.8150*CCO+28.8112*CCO**2-29.9257*CCO BETA2490
   1**3+11.6815*CCO**4 $ HHA=COREFA * HSHOL                          BETA2500
  7 RETURN                                                           BETA2510
    END                                                              BETA2520
```

Example of Input

X	Y	B1	B2	T	A	NOR	DIST	TIME	GRID	MM	NN
4500.	33000.	1.	1.	20.	-35.	18	1500.	30.	1500.	52	56

440.	460.	515.	505.	510.	470.	425.	355.	300.	330.	370.	360.	340.	330.	-11
340.	345.	350.	365.	375.	390.	455.	560.	700.	850.	950.	865.	705.	630.	-10
550.	490.	470.	460.	415.	380.	335.	400.	460.	520.	560.	595.	620.	590.	-9
510.	460.	400.	315.	275.	300.	220.	140.	80.	55.	53.	51.	51.	51.	-8
360.	405.	450.	435.	445.	380.	330.	270.	235.	250.	280.	280.	260.	270.	-7
290.	297.	300.	305.	315.	360.	420.	490.	655.	810.	930.	870.	750.	620.	-6
540.	500.	515.	515.	480.	445.	440.	500.	540.	593.	630.	630.	600.	590.	-5
610.	560.	515.	400.	315.	275.	234.	190.	110.	63.	53.	50.	50.	50.	-4
295.	370.	370.	325.	325.	307.	260.	190.	170.	175.	190.	190.	180.	205.	-3
240.	250.	250.	265.	275.	305.	405.	470.	570.	750.	900.	885.	800.	630.	-2
560.	540.	550.	560.	550.	540.	550.	605.	620.	640.	610.	570.	530.	510.	-1
540.	590.	570.	430.	300.	310.	280.	240.	170.	70.	55.	50.	50.	50.	0
240.	250.	235.	195.	225.	200.	190.	160.	130.	140.	160.	170.	170.	180.	1
190.	195.	210.	230.	250.	280.	350.	427.	490.	670.	820.	860.	820.	680.	2
610.	590.	590.	615.	600.	610.	620.	660.	650.	620.	560.	505.	480.	470.	3
480.	520.	560.	515.	405.	370.	320.	260.	190.	80.	60.	49.	49.	49.	4
150.	195.	200.	110.	125.	120.	115.	95.	80.	98.	105.	110.	115.	135.	5
162.	170.	180.	205.	210.	210.	290.	380.	455.	580.	740.	785.	840.	720.	6
660.	650.	640.	660.	660.	670.	690.	685.	650.	585.	510.	480.	465.	440.	7
450.	490.	550.	525.	450.	380.	325.	270.	215.	120.	65.	50.	50.	50.	8
80.	80.	90.	75.	73.	70.	72.	70.	70.	70.	72.	80.	85.	95.	9
110.	120.	140.	160.	175.	195.	240.	290.	390.	475.	610.	670.	750.	815.	10
760.	720.	715.	725.	730.	710.	680.	655.	620.	540.	480.	445.	430.	420.	11
440.	410.	550.	490.	415.	370.	325.	285.	235.	170.	85.	65.	50.	50.	12
61.	60.	60.	58.	60.	60.	61.	63.	60.	60.	65.	65.	75.	75.	13
80.	85.	100.	120.	130.	120.	160.	230.	280.	380.	460.	540.	630.	775.	14
770.	750.	735.	715.	665.	635.	620.	610.	570.	485.	440.	415.	385.	375.	15
425.	490.	550.	500.	445.	375.	315.	295.	245.	190.	110.	80.	65.	50.	16
47.	47.	48.	49.	52.	54.	53.	54.	56.	59.	60.	60.	60.	62.	17
63.	65.	70.	77.	82.	85.	110.	150.	185.	250.	340.	395.	465.	550.	18
560.	610.	590.	600.	570.	540.	525.	520.	500.	460.	410.	380.	360.	365.	19
420.	510.	515.	500.	465.	410.	283.	265.	235.	210.	145.	105.	60.	52.	20
38.	38.	38.	41.	46.	48.	51.	53.	54.	56.	57.	57.	59.	60.	21
60.	62.	63.	67.	70.	70.	80.	95.	115.	160.	210.	300.	370.	450.	22
450.	465.	490.	510.	470.	465.	460.	460.	440.	415.	395.	385.	370.	360.	23
430.	520.	470.	430.	415.	420.	240.	270.	254.	260.	250.	215.	160.	55.	24
31.	32.	33.	38.	40.	43.	48.	50.	51.	54.	55.	56.	57.	58.	25
59.	60.	61.	62.	63.	64.	65.	70.	80.	86.	140.	185.	285.	385.	26
360.	380.	420.	415.	400.	390.	390.	410.	380.	365.	350.	350.	350.	352.	27
500.	470.	410.	360.	350.	370.	380.	375.	340.	310.	315.	300.	195.	120.	28
30.	28.	33.	36.	35.	35.	38.	43.	46.	49.	51.	53.	55.	56.	29
58.	59.	60.	60.	62.	62.	63.	64.	65.	73.	80.	120.	185.	260.	30
240.	280.	325.	325.	325.	330.	340.	330.	310.	290.	305.	320.	405.	450.	31
460.	410.	300.	260.	260.	295.	300.	290.	300.	300.	290.	310.	300.	220.	32
13.	24.	26.	28.	28.	31.	34.	38.	42.	45.	48.	51.	54.	54.	33
56.	57.	58.	59.	59.	60.	60.	62.	64.	66.	70.	120.	160.	160.	34
155.	180.	240.	240.	230.	230.	260.	270.	270.	280.	325.	370.	440.	470.	35
420.	370.	300.	240.	240.	210.	180.	200.	200.	190.	210.	220.	300.	260.	36
18.	20.	24.	25.	25.	26.	27.	33.	36.	40.	44.	49.	51.	53.	37
55.	56.	58.	58.	57.	59.	60.	60.	62.	63.	63.	85.	120.	105.	38

Example of Output

XV	YV	B1	B2	T	A1	NOR	DIST	TIME	GRID	MM	NN
4500.0	33000.0	1	1	20	-35	18	1500.00	30.0	1500.0	52	56

NUMBER OF CREST = 1

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
2339.60	10412.11	1.000	1.000	2	2626.38	10821.68	1.000	1.000	2
2913.17	11231.26	1.000	1.000	2	3199.96	11640.84	1.000	1.000	2
3486.75	12050.41	1.000	1.000	2	3773.54	12459.99	1.000	1.000	2
4060.33	12869.56	1.000	1.000	2	4347.11	13279.14	1.000	1.000	2
4633.90	13688.72	1.000	1.000	2	4920.69	14098.29	1.000	1.000	2
5207.48	14507.87	1.000	1.000	2	5494.27	14917.44	1.000	1.000	2
5781.05	15327.02	1.000	1.000	2	6067.84	15736.60	1.000	1.000	2
6354.63	16146.17	1.000	1.000	2	6641.42	16555.75	1.000	1.000	2
6928.21	16965.33	1.000	1.000	2	7215.00	17374.90	1.000	1.000	2

NUMBER OF CREST = 2

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
3179.19	9824.22	1.000	1.000	2	3465.98	10233.79	1.000	1.000	2
3752.77	10643.37	1.000	1.000	2	4039.56	11052.94	1.000	1.000	2
4326.35	11462.52	1.000	1.000	2	4613.13	11872.10	1.000	1.000	2
4899.92	12281.67	1.000	1.000	2	5186.71	12691.25	1.000	1.000	2
5473.50	13100.83	1.000	1.000	2	5760.29	13510.40	1.000	1.000	2
6047.07	13919.98	1.000	1.000	2	6333.86	14329.55	1.000	1.000	2
6620.65	14739.13	1.000	1.000	2	6913.46	15164.19	1.000	.905	2
7182.28	15566.65	1.000	.899	2	7454.38	15986.51	1.000	.897	2
7728.66	16404.84	1.000	.895	2	8001.84	16823.95	1.000	.895	2

NUMBER OF CREST = 3

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
4005.32	9245.76	1.000	.902	2	4292.80	9654.85	1.000	.899	2
4577.71	10065.74	1.000	.897	2	4858.41	10479.58	1.000	.897	2
5140.28	10890.72	1.000	.897	2	5423.42	11304.73	1.000	.898	2
5711.14	11713.66	1.000	.898	2	6001.92	12096.33	1.000	.898	2
6307.35	12513.13	1.000	.903	2	6587.17	12931.41	1.000	.903	2
6854.59	13354.55	1.000	.898	2	7124.80	13775.74	1.000	.896	2
7394.65	14197.17	1.000	.895	2	7694.29	14666.16	1.024	.917	2
7930.90	15056.20	1.004	.899	2	8191.41	15480.48	1.032	.923	2
8446.16	15904.45	1.031	.922	2	8727.76	16319.95	1.009	.903	2

NUMBER OF CREST = 22

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
10264.28	4616.24	0	0	1	10853.02	4550.71	0	0	1
11438.10	4352.73	0	0	1	12072.87	3316.63	0	0	1
12920.26	2737.33	0	0	1	15666.05	1290.81	.270	.291	2
17781.43	2925.94	.467	.486	2	16546.11	1779.73	8.942	9.377	2
18531.25	4023.30	.882	.924	2	18968.07	4701.04	.769	.805	2
19101.90	4934.51	.757	.791	2	19511.47	5662.92	.859	.900	2
19789.68	6181.58	1.139	1.199	2	20449.56	7333.27	1.204	1.271	2
20588.90	7626.71	.930	.986	2	20819.12	8054.09	2.473	2.659	2
20812.04	8068.66	1.437	1.542	2	21047.31	8517.89	1.045	1.125	2

NUMBER OF CREST = 23

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
10264.28	4616.24	0	0	1	10853.02	4550.71	0	0	1
11438.10	4352.73	0	0	1	12072.87	3316.63	0	0	1
12920.26	2737.33	0	0	1	15818.45	860.81	.252	.287	2
18177.11	2611.34	.438	.488	2	16808.11	1362.17	6.696	7.344	2
18949.52	3760.33	.882	.962	2	19386.59	4438.11	.756	.840	2
19523.12	4669.33	.725	.807	2	19940.13	5417.85	.850	.949	2
20218.44	5951.79	1.143	1.270	2	20883.05	7120.80	1.131	1.285	2
21014.87	7416.37	.942	1.068	2	21227.05	7848.73	2.254	2.625	2
21212.27	7844.86	1.365	1.584	2	21454.05	8311.97	1.031	1.228	2

NUMBER OF CREST = 24

X	Y	COREFR	HHO	NGO	X	Y	COREFR	HHO	NGO
10264.28	4616.24	0	0	1	10853.02	4550.71	0	0	1
11438.10	4352.73	0	0	1	12072.87	3316.63	0	0	1
12920.26	2737.33	0	0	1	15936.15	482.73	.238	.296	2
18502.09	2350.98	.415	.536	2	17039.75	997.51	5.649	7.103	2
19320.29	3526.00	.863	1.084	2	19743.52	4220.11	.753	.964	2
19879.31	4454.54	.708	.915	2	20297.47	5212.49	.843	1.131	2
20591.71	5763.85	1.124	1.408	2	21240.91	6950.72	1.088	1.493	2
21375.95	7248.08	.945	1.397	2	21569.75	7698.52	2.148	3.043	2
21548.37	7673.71	1.332	1.864	2	21784.38	8176.83	1.041	1.774	2

APPENDIX II

```
      SUBROUTINE DRAW (NUMPTS, X, Y, MODCURV, ITYPE, LABEL,
     1                ITITLE, EXSCALE, YSCALE, IXUP, IYRIGHT,
     2                 MODEXAX, MODEYAX, IWIDE, IHIGH, IGRID,
     3                LAST)
C
C
C     A GENERAL CURVE DRAWING AND POINT PLOTTING SUBROUTINE
C         PROGRAMMER    J. R. WARD
C         DATE          FEB. 1964, REVISED JUNE 1965
C         SYSTEM        FORTRAN 60
C         OUTPUT        LOGICAL TAPE NUMBER 8
C         NOTE          ASTERISKS MARK CHANGES FOR FORTRAN 63
C
C     INPUT ARGUMENTS ---
C
C         1. NUMPTS       NUMBER OF POINTS TO BE PLOTTED. THIS MUST ALWAYS
C                         BE AT LEAST 2, AND MUST NOT EXCEED 30 FOR POINT
C                         PLOTTING, OR 900 FOR CURVE DRAWING.
C
C         2. X            ARRAY OF X-ORDINATES. DIMENSION AT LEAST EQUAL
C                         TO NUMPTS AND NOT MORE THAN 900 IN CALLING
C                         PROGRAM.
C
C         3. Y            ARRAY OF Y-ORDINATES. DIMENSION AS FOR THE
C                         X ARRAY IN THE CALLING PROGRAM.
C
C         4. MODCURV      CONTROLS THE NUMBER OF CURVES, AND/OR SETS OF
C                         POINTS, ON EACH GRAPH. THE CODES ARE
C                                 0     ONLY ONE PLOT ON THIS GRAPH
C                                 1     FIRST PLOT ON MULTI-PLOT GRAPH
C                                 2     INTERMEDIATE PLOT ON MULTI-PLOT GRAPH
C                                 3     LAST PLOT ON MULTI-PLOT GRAPH.
C
C         5. ITYPE        CONTROLS THE TYPE OF PLOT. THE CODES ARE
C                                 0     STRAIGHT LINES JOIN SUCCESSIVE POINTS
C                                       (STANDARD CURVE DRAWING)
C                                 1     POINTS PLOTTED WITH CROSS (X)
C                                 2     POINTS PLOTTED WITH PLUS (+)
C                                 3     POINTS PLOTTED WITH SQUARE
C                                 4     POINTS PLOTTED WITH DIAMOND
C                                 5     POINTS PLOTTED WITH TRIANGLE
C                         WHEN POINTS ARE BEING PLOTTED (ITYPE=1 THRU. 5),
C                         THE POINTS ARE NOT CONNECTED.
C
C         6. LABEL        IF A CURVE IS BEING DRAWN (ITYPE = 0), LABEL IS
C                         A 4-CHARACTER BCD CURVE IDENTIFIER WHICH WILL BE
C                         REPRODUCED AT THE END OF THE CURVE. FOR EXAMPLE,
C                         LABEL = 4H ONE . IF POINTS ARE BEING PLOTTED,
C                         LABEL IS AN 8-CHARACTER IDENTIFIER. THE FIRST 4
C                         CHARACTERS ARE REPRODUCED WITH THE FIRST PLOTTED
```

```
C                    POINT, AND THE LAST 4 WITH THE LAST POINT. SET
C                    TO BLANK ANY UNWANTED CHARACTERS.
C
C      7. ITITLE     AN ARRAY OF TWELVE 8-CHARACTER BCD WORDS.
C                    THE FIRST SIX WORDS WILL BE REPRODUCED AS THE
C                    FIRST LINE OF GRAPH TITLE, AND THE LAST SIX
C                    WORDS WILL FORM THE SECOND LINE. THE TITLE MUST
C                    INCLUDE THE USERS JOB IDENTIFICATION. DIMENSION
C                    12 IN CALLING PROGRAM, AND SET TO BLANK ALL
C                    UNWANTED CHARACTERS.
C
C      8. EXSCALE    X-SCALE (UNITS/INCH) AS POSITIVE FLOATING POINT
C                    VARIABLE WITH ONE FIGURE SIGNIFICANCE. SET TO
C                    ZERO FOR AUTO-SCALE.
C
C      9. YSCALE     Y-SCALE (UNITS/INCH) AS POSITIVE FLOATING POINT
C                    VARIABLE WITH ONE FIGURE SIGNIFICANCE. SET TO
C                    ZERO FOR AUTO-SCALE.
C
C     10. IXUP       X-AXIS OFFSET FROM BOTTOM OF GRAPH IN INCHES.
C                    THIS MUST NOT EXCEED IHIGH, AND MUST NOT BE
C                    NEGATIVE.
C
C     11. IYRIGHT    Y-AXIS OFFSET FROM LEFT OF GRAPH IN INCHES. THIS
C                    MUST NOT EXCEED IWIDE, AND MUST NOT BE NEGATIVE.
C
C     12. MODEXAX    MODE OF X-AXIS OFFSET. SEE CODES BELOW.
C
C     13. MODEYAX    MODE OF Y-AXIS OFFSET. THE CODES ARE AS FOLLOWS
C                            0    COMPUTED OFFSET, HOLDING ORIGIN ON
C                                 GRAPH. THE CORRESPONDING IXUP OR
C                                 IYRIGHT IS IGNORED
C                            1    COMPUTED OFFSET, WITH ORIGIN OFF THE
C                                 GRAPH IF APPROPRIATE. THE CORRESPOND-
C                                 ING IXUP OR IYRIGHT IS IGNORED. USE
C                                 ONLY WITH AUTO-SCALE
C                            2    AXIS OFFSET AS SPECIFIED BY IXUP OR
C                                 IYRIGHT.
C
C     14. IWIDE      WIDTH OF GRAPH IN INCHES. THIS MUST NOT EXCEED
C                    NINE.  ZERO WILL BE READ AS EIGHT INCHES.
C
C     15. IHIGH      HEIGHT OF GRAPH IN INCHES. THIS MUST NOT EXCEED
C                    FIFTEEN. ZERO WILL BE READ AS EIGHT INCHES.
C
C     16. IGRID      IF SET TO 1, A ONE INCH BY ONE INCH GRID WILL
C                    BE SUPERIMPOSED ON THE GRAPH.
C
C     17. LAST       INDICATES TO CALLING PROGRAM WHETHER LAST PLOT
C                    WAS COMPLETED SUCCESSFULLY. THE CODES ARE
C                            0    LAST PLOT COMPLETED SUCCESSFULLY
C                            1    LAST PLOT NOT SUCCESSFUL
C                            2    LAST PLOT NOT SUCCESSFUL AND NO
```

```
C                                     FURTHER GRAPH OUTPUT WILL BE ATTEMPT-
C                                     ED UNTIL MODCURV IS NEXT ONE OR ZERO
C                                3    DRAW WAS ENTERED WITH MODCURV NOT
C                                     EQUAL TO ONE OR ZERO WHILE THE ERROR
C                                     LOCK-OUT WAS SET.
C                          THIS ARGUMENT MUST ALWAYS BE A NAME IN THE CALL
C                          STATEMENT. NEVER A NUMBER.
C
C
C     NOTE ---
C         ALL ARGUMENTS FROM NUMBER 7 THRU. NUMBER 16 ARE IGNORED WHEN
C         MODCURV IS EITHER 2 OR 3. HOWEVER, ARGUMENTS MUST NEVER BE
C         OMITTED FROM THE CALLING STATEMENT. IT IS MERELY THEIR VALUES
C         WHICH ARE THEN IRRELEVANT. ARGUMENTS MAY BE LISTED BY NAME OR
C         VALUE IN THE CALL STATEMENT. NO VALUE IN THE CALLING PROGRAM
C         WILL BE ALTERED BY THIS SUBROUTINE.
C
C     REFERENCE ---
C         THE BINARY TAPE FORMAT REQUIRED BY THE OFF-LINE PLOTTER IS
C         DESCRIBED IN THE WRITEUP OF THE CDC 160A GRAPH PLOT PROGRAM
C         (IDENT. B001). THE FORMAT REQUIRED BY THE CDC 160 PROGRAM IS
C         SIMILAR EXCEPT THAT THE INTERPOLATION ARGUMENT MUST BE ZERO.
C
C
      DIMENSION X(900), Y(900), ITITLE(12), JXTIT(12), JYTIT(12),
     1          LTITLE(14), KAXIS(5), ICURV(460), JGRID(25), ICONT(1),
     2          JJTITLE(12)
      IPOINT = ITYPE
      CON(           ICONTRL = 40000B).
      CON(ICURV3 = 3777377720202020B, ICURV4 = 0104000000000000B).
*         REPLACE WITH DATA STATEMENT IN FORTRAN 62-3.
*        PUT ITEST = 0 IN DATA STATEMENT.
      IF (JTEST - 73546912)  9070,9071,9070
 9070 ITEST = 0
      JTEST = 73546912
 9071 CONTINUE
*        REMOVE ABOVE NONSENSE IN FORTRAN 63.
C         CHECK PREVIOUS OPERATION OF ROUTINE, IF ANY. CODES ARE
C             ITEST = 0    IF PREVIOUS GRAPH, IF ANY, COMPLETED
C             ITEST = 1    IF PREVIOUS GRAPH NOT COMPLETED
C             ITEST = 2    IF ERROR FOUND WHILE MODCURV WAS ONE, OR IF
C                          MODCURV WAS ILLEGAL.
      IF(ITEST - 2)1000,1001,1000
 1001 IF(MODCURV)1003,1002,1003
 1002 ITEST = 0
      GO TO 1000
 1003 IF(MODCURV - 1)1004,1002,1004
 1004 LAST = 3
      RETURN
C         SET UP ERROR RETURN ROUTINE. ENTRY AT STATEMENT 1005.
 1005 IF(ITEST)1009,1006,1009
 1006 IF(MODCURV)1007,1008,1007
 1007 PRINT 1100
```

```
 1100 FORMAT ( 59H NO FURTHER GRAPH OUTPUT UNTIL MODCURV NEXT IS ZERO OR
     1 ONE. ,/)
      ITEST = 2
      LAST = 2
      RETURN
 1008 PRINT 1101
 1101 FORMAT ( 30H THIS PLOT WILL NOT BE OUTPUT. ,/)
      LAST = 1
      RETURN
 1009 IF(MODCURV - 2)1010,1008,1010
 1010 IF(MODCURV - 3)1007,1011,1007
 1011 ITEST = 0
      GO TO 1008
C         CHECK LEGALITY OF INPUT ARGUMENTS.
 1000 IF(NUMPTS - 2)1,2,2
    1 PRINT 100
  100 FORMAT (/, 32H NUMPTS MUST NOT BE LESS THAN 2. )
      GO TO 1005
    2 IF(IPOINT)9000,9004,9001
 9000 PRINT 9100
 9100 FORMAT (/, 15H ILLEGAL ITYPE. )
      GO TO 1005
 9001 IF(IPOINT - 5)9002,9002,9000
 9002 IF(NUMPTS - 30)3,3,9003
 9003 PRINT 9101
 9101 FORMAT (/, 46H NUMPTS MUST NOT EXCEED 30 FOR POINT PLOTTING. )
      GO TO 1005
 9004 IF(NUMPTS - 900)3,3,9005
 9005 PRINT 9102
 9102 FORMAT (/, 28H NUMPTS MUST NOT EXCEED 900. )
      GO TO 1005
    3 IX = 1HX
      IY = 1HY
      AMAXX = -0.2E+100
      AMAXY = -0.2E+100
      AMINX = +0.2E+100
      AMINY = +0.2E+100
      DO 1020 I= 1,NUMPTS
      AMAXX = MAX1F(X(I),AMAXX)
      AMAXY = MAX1F(Y(I),AMAXY)
      AMINX = MIN1F(X(I),AMINX)
 1020 AMINY = MIN1F(Y(I),AMINY)
      AMAXA = MAX1F(ABSF(AMAXX), ABSF(AMAXY), ABSF(AMINX), ABSF(AMINY))
      IF(AMAXA - 1.0E+99)1022,1022,1021
 1021 PRINT 1102
 1102 FORMAT (/, 58H NO X OR Y VALUE MAY EXCEED 1.0E+99 IN ABSOLUTE MAGN
     1ITUDE. )
      GO TO 1005
 1022 IF(ABSF(AMAXX - AMINX) - 1.0E-97)1023,1025,1025
 1023 IF(ABSF(AMAXY - AMINY) - 1.0E-97)1024,1025,1025
 1024 PRINT 1103
 1103 FORMAT (/, 38H ALL POINTS HAVE THE SAME COORDINATES. )
      GO TO 1005
```

```
1025 IF(ITEST)4,7,4
   4 IF(MODCURV - 2)5,240,5
   5 IF(MODCURV - 3)6,240,6
   6 PRINT 101
 101 FORMAT (/, 17H ILLEGAL MODCURV. )
     GO TO 1005
   7 IF(MODCURV)6,9,8
   8 IF(MODCURV - 1)6,9,6
   9 IF(IWIDE)10,11,12
  10 ITIT = 5HIWIDE
     PRINT 102, ITIT, ITIT
 102 FORMAT (/,  9H ILLEGAL ,A5,29H. GRAPH WILL BE PLOTTED WITH ,A5,
    1            5H = 8. ,/)
  11 JWIDE = 8
     GO TO 14
  12 IF(IWIDE - 9)13,13,10
  13 JWIDE = IWIDE
  14 IF(IHIGH)15,16,17
  15 ITIT = 5HIHIGH
     PRINT 102, ITIT, ITIT
  16 JHIGH = 8
     GO TO 19
  17 IF(IHIGH - 15)18,18,15
  18 JHIGH = IHIGH
  19 NODEXAX = MODEXAX
     IF(MODEXAX)20,27,21
  20 ITIT= 8HMODEXAX.
     PRINT 104, ITIT, IX
 104 FORMAT (/,  9H ILLEGAL ,A8, 32H GRAPH WILL BE PLOTTED WITH MODE,
    1         A1,  7HAX = 0. ,/)
     NODEXAX = 0
     GO TO 27
  21 IF(MODEXAX - 2)27,22,20
  22 IF(IXUP - JHIGH)24,24,23
  23 ITIT = 8HIXUP.
     PRINT 104, ITIT, IX
     NODEXAX = 0
     GO TO 27
  24 IF(IXUP)23,26,26
  26 JXUP = IXUP
  27 NODEYAX = MODEYAX
     IF(MODEYAX)28,35,29
  28 ITIT=8HMODEYAX.
     PRINT 104, ITIT, IY
     NODEYAX = 0
     GO TO 35
  29 IF(MODEYAX - 2)35,30,28
  30 IF(IYRIGHT - JWIDE)32,32,31
  31 ITIT = 8HIYRIGHT.
     PRINT 104, ITIT, IY
     NODEYAX = 0
     GO TO 35
  32 IF(IYRIGHT) 31,34,34
```

```
   34 JYRIGHT = IYRIGHT
C         INITIALIZE PRIOR TO SCALING AND AXIS LOCATING.
C         IFLAG = 0 FOR PASS WITH XDATA. IFLAG = 1 FOR PASS WITH YDATA.
   35 DO 2235 IOTA=1,12
 2235 JJTITLE(IOTA) = ITITLE(IOTA)
      IFLAG = 0
      BETA = 0.
      SCALE = EXSCALE
      IAXIS = JYRIGHT
      MODE = NODEYAX
      ISIZE = JWIDE
      IXY = IX
      IYX = IY
      AMAX = AMAXX
      AMIN = AMINX
      GO TO 52
   50 IFLAG = 1
      BETA = 0.
      SCALE = YSCALE
      IAXIS = JXUP
      MODE  = NODEXAX
      ISIZE = JHIGH
      AMAX = AMAXY
      AMIN = AMINY
      IXY = IY
      IYX  = IX
C         CHECK SCALE AND GO TO FIXED OR AUTO SCALE ROUTINES.
   52 IF(SCALE)53,59,56
   53 PRINT 114, IXY, IXY
  114 FORMAT (/,  9H ILLEGAL ,A1,39HSCALE. GRAPH WILL BE PLOTTED WITH AU
     1TO ,A1, 7H-SCALE. ,/)
      GO TO 59
C     EXPRESS FIXED SCALE IN E FORMAT WITH ONE FIGURE SIGNIFICANCE.
   56 IF(SCALE - 1.0E+99)57,53,53
   57 IF(SCALE - 1.0E-99)53,53,58
   58 CALL SCALEIT(SCALE,ISCAL10,FACTOR,1)
      SCALE = FACTOR*10.**ISCAL10
C         CHECK AND COMPUTE AXIS LOCATION IF NECESSARY. FIXED SCALE
C         CASE. ITAG = 0 IF ORIGIN ON GRAPH OR 1 IF IT IS SUPPRESSED.
      IF(MODE - 1)1032,1031,1030
 1030 ITAG =0
      GO TO 203
 1031 PRINT 1104 , IYX, IXY, IXY
 1104 FORMAT (/, 5H MODE,A1,24HAX MUST NOT BE 1 UNLESS ,A1,57HSCALE IS 
     1 (AUTO-SCALE). GRAPH WILL BE PLOTTED WITH AUTO ,A1, 7H-SCALE. ,/)
      GO TO 59
 1032 IF(ABSF(AMAX - AMIN) - 1.0E-97)1033,1038,1038
 1033 IF(ABSF(AMAX) - 1.0E-97)1034,1039,1039
 1039 IF(AMAX)1036,1034,1037
 1034 IAXIS = ISIZE/2
      GO TO 1030
 1036 IAXIS = ISIZE
      GO TO 1030
```

```
1037 IAXIS = 0
     GO TO 1030
1038 IF(SIGNF(1.,AMAX) - SIGNF(1.,AMIN))1040,1039,1040
1040 ASIZE = ISIZE
     IAXIS = -AMIN/(AMAX - AMIN)*ASIZE +0.5
     GO TO 1030
C          AUTO SCALE ROUTINE.
  59 IF(MODE - 1)60,64,69
  60 AMAX = MAX1F(0., AMAX)
     AMIN = MIN1F(0., AMIN)
  64 IF(ABSF(AMAX - AMIN) - 1.0E-97)65,68,68
  65 PRINT 116, IXY, IXY, IYX
 116 FORMAT (/,  5H ALL ,A1,47H VALUES EQUAL. AUTO SCALE POSSIBLE ONLY
    1IF THE ,A1,29H VALUES ARE NON-ZERO AND MODE,A1, 7HAX = 2. )
     GO TO 1005
  68 ASIZE = ISIZE
     SCALE = (AMAX - AMIN)/ASIZE
     GO TO 83
  69 IF(ABSF(AMAX - AMIN) - 1.0E-97)70,74,74
  70 IF(ABSF(AMAX) - 1.0E-97)71,74,74
  71 PRINT 118, IXY
 118 FORMAT (/,  5H ALL ,A1,38H VALUES ZERO. AUTO SCALE NOT POSSIBLE. )
     GO TO 1005
  74 IF(AMAX - 1.0E-97) 76,75,75
  75 IF(ISIZE - IAXIS)77,76,77
  76 SCALE1 = 0.
     GO TO 78
  77 AXIS = IAXIS
     ASIZE = ISIZE
     SCALE1 = AMAX/(ASIZE - AXIS)
  78 IF(AMIN + 1.0E-97)79,79,80
  79 IF(IAXIS)81,80,81
  80 SCALE2 = 0.
     GO TO 82
  81 AXIS = IAXIS
     SCALE2 = -AMIN/AXIS
  82 IF(SCALE1 + SCALE2)1984,1982,1984
1982 PRINT 1983, IYX, IYX
1983 FORMAT (/, 56H NONE OF THE PLOT LIES ON THE GRAPH WITH THIS SPECIF
    1IED ,A1,47H-AXIS LOCATION. GRAPH WILL BE PLOTTED WITH MODE,A1,
    2  7HAX = 0.  ,/)
     MODE = 0
     GO TO 60
1984 SCALE = MAX1F(SCALE1, SCALE2)
  83 CALL SCALEIT(SCALE, ISCAL10,FACTOR,3)
     IF(FACTOR - 5.05)85,85,84
  84 FACTOR = 1
     ISCAL10 = ISCAL10 + 1
     GO TO 90
  85 IF(FACTOR - 2.02)87,87,86
  86 FACTOR = 5
     GO TO 90
  87 IF(FACTOR - 1.01)89,89,88
```

```
   88 FACTOR = 2
      GO TO 90
   89 FACTOR = 1
   90 SCALE = FACTOR*10.**ISCAL10
C         COMPUTE AXIS LOCATION IF NECESSARY. AUTO SCALE CASE.
      IF(MODE - 1)92,91,93
   91 IF(SIGNF(1.,AMAX) - SIGNF(1.,AMIN))92,94,92
   92 IAXIS = -AMIN/SCALE + 0.5
   93 ITAG = 0
      GO TO 203
   94 IF(AMAX)95,95,200
   95 IAXIS = ISIZE
      BETA = -AMAX/SCALE
      IF(BETA - 1.E+12)99,99,96
   96 PRINT 120, IXY
  120 FORMAT (/, 15H THE ORIGIN OF ,A1, 43H CANNOT BE OFFSET MORE THAN 1
     1.0E+12 INCHES. )
      GO TO 1005
   99 IBETA = BETA + 0.5
      BETA = -IBETA
C         BETA IS THE NUMBER OF INCHES OF ORIGIN SUPPRESSION,POSITIVE IF
C         TRUE ORIGIN IS BELOW OR TO LEFT OF THE GRAPH.
      IF(BETA + 1.)97,97,93
   97 ITAG = 1
      GO TO 203
  200 IAXIS = 0
      BETA = AMIN/SCALE
      IF(BETA - 1.E+12)201,201,96
  201 IBETA = BETA + 0.5
      BETA = IBETA
      IF(BETA - 1.)93,202,202
  202 ITAG = 1
C         RELEASE RESULTS TO REMAINING PART OF PROGRAM. START SECOND
C         PASS FOR Y VALUES IF NOT YET COMPUTED.
  203 IF(IFLAG)205,204,205
  204 SCALEX = SCALE
      IXFACT  = FACTOR
      IXSC10 = ISCAL10
      IXAXIS = IAXIS
      ITAGX  = ITAG
      ISIZEX = ISIZE
      BETAX = BETA
      GO TO 50
  205 BETAY = BETA
      SCALEY = SCALE
      IYFACT =  FACTOR
      IYSC10 = ISCAL10
      IYAXIS = IAXIS
C         NOW WRITE RECORDS.
      ITAGY  = ITAG
      ISIZEY = ISIZE
C         THIS COMPLETES CALCULATION OF SCALE FACTORS ETC. NOW GENERATE
C         TAPE RECORDS. FIRST, THE SCALE FACTOR TITLES.
```

```
  206 JXTIT(1) = 8H1     X-
      JXTIT(2) = 8HSCALE =
      JXTIT(3) = ICODE(SCALEX)
      JXTIT(4) = 8H UNITS/I
      JXTIT(5) = 8HNCH.
      JYTIT(1) = 8H1     Y-
      JYTIT(2) = 8HSCALE =
      JYTIT(3) = ICODE(SCALEY)
      JYTIT(4) = 8H UNITS/I
      JYTIT(5) = 8HNCH.
      DO 9206 I=6,11
      JXTIT(I) = 8H
 9206 JYTIT(I) = 8H
      IF(ITAGX)211,211,207
  207 IF(BETAX)208,208,209
  208 JXTIT(7) = 8H   ADD -
      GO TO 210
  209 JXTIT(7) = 8H   ADD +
  210 JXTIT(8) = ICODE (BETAX*SCALEX)
      JXTIT(9) = 8H UNITS T
      JXTIT(10)= 8HO ALL X
      JXTIT(11)= 8HVALUES.
  211 IF(ITAGY)216,216,212
  212 IF(BETAY)213,213,214
  213 JYTIT(7) = 8H   ADD -
      GO TO 215
  214 JYTIT(7) = 8H   ADD +
  215 JYTIT(8) = ICODE (BETAY*SCALEY)
      JYTIT(9) = 8H UNITS T
      JYTIT(10)=8HO ALL Y
      JYTIT(11)= 8HVALUES.
  216 ICONT(1) = ICONTRL + 4
C         INSERT TITLE SIZE (02B) AHEAD OF MAIN TITLE RECORD.
      CALL ISHIFT6 (ITITLE, LTITLE)
C        TEST FOR ALL BLANK TITLES.
      ICHECK = 8H
      DO 9075 I=1,6
      IF(ITITLE(I)-ICHECK) 9074,9075,9074
 9074 IF(ITITLE(I) ) 9080,9075,9080
 9075 CONTINUE
      IT1 = 1
      ICONT(1) = ICONT(1) - 1
      GO TO 9081
 9080 IT1 = 0
 9081 DO 9085 I=7,12
      IF (ITITLE(I) - ICHECK) 9084,9085,9084
 9084 IF (ITITLE(I))9090,9085,9090
 9085 CONTINUE
      IT2 = 1
      ICONT(1) = ICONT(1) - 1
      GO TO 9091
 9090 IT2 = 0
C         NOW GENERATE AXES RECORDS.
```

```
 9091 LFTMGN = 0*100
      IBOTMGN = 1*100
      IH = LFTMGN
      JH = IBOTMGN +IYAXIS*100
      LH = ISIZEX*100
      IHL = LFTMGN + ISIZEX*100 - 107
      KAXIS(1) = IPACK12(IH,JH,LH,IHL)
      JHL = JH - 13
      IHL2 = -100
      IVH  = (ISIZEX -IXAXIS - 1)*IXFACT
      IVH2 = -IXFACT
      KAXIS(2) = IPACK12(JHL,IHL2,IVH,IVH2)
      NH = ISIZEX
      ISH = 8H      14
      IV = LFTMGN + IXAXIS*100
      JV = IBOTMGN
      KAXIS(3) = IPACK12(NH,ISH,IV,JV)
      LV = ISIZEY*100
      IVL = IV - 3
      JVL = IBOTMGN + ISIZEY*100 - 107
      JVL2 = -100
      KAXIS(4) = IPACK12(LV,IVL,JVL,JVL2)
      IVV = (ISIZEY - IYAXIS - 1)*IYFACT
      IVV2 = -IYFACT
      INV = ISIZEY
      ISV = 8H      11
      KAXIS(5) = IPACK12(IVV,IVV2,INV,ISV)
C         NOW GENERATE CURVES.
      SCX = 100./SCALEX
      SCY = 100./SCALEY
      EXAXIS= IXAXIS*100
      YAXIS = IYAXIS*100
      ALFTMGN = LFTMGN
      BOTMGN = IBOTMGN
      SHIFTX =EXAXIS - BETAX*100. + ALFTMGN
      SHIFTY = YAXIS - BETAY*100. + BOTMGN
      EXSIZE= ISIZEX*100 + LFTMGN + 60
      SIZEX = LFTMGN - 60
      YSIZE = ISIZEY*100 + IBOTMGN + 70
      SIZEY = IBOTMGN - 70
      ICURV(1) = 0
  240 IF(IPOINT)9010,9007,9010
 9007 IF(XMODF(NUMPTS,2))9700,9701,9700
 9700 ISWITCH = 1
      GO TO 242
 9701 ISWITCH = 2
  242 INUM = (NUMPTS + 1)/2
      DO 244 I=1,INUM
      C1 = X(2*I-1)*SCX + SHIFTX
      C2 = Y(2*I-1)*SCY + SHIFTY
      IF(I-INUM)241,9241,241
 9241 GO TO (9242,241),ISWITCH
 9242 C3 = C1
```

```
      C4 = C2
      GO TO 9243
  241 C3 = X(2*I)*SCX + SHIFTX
      C4 = Y(2*I)*SCY + SHIFTY
 9243 C1 = MIN1F(C1,EXSIZE)
      IC1= MAX1F(C1, SIZEX)
      C2 = MIN1F(C2, YSIZE)
      IC2= MAX1F(C2, SIZEY)
      C3 = MIN1F(C3,EXSIZE)
      IC3= MAX1F(C3, SIZEX)
      C4 = MIN1F(C4, YSIZE)
      IC4= MAX1F(C4, SIZEY)
  244 ICURV(I+1) = IPACK12(IC1,IC2,IC3,IC4)
      II = INUM + 3
  246 CALL IPACKL1(LABEL, LABEL1, IDUMMY)
      ICURV(II-1) = LABEL1
      ICURV(II) = ICURV4
 9010 IF(MODCURV - 1)247,247,9015
  247 CALL IREADY (IDUMMY)
      IF(IDUMMY)5000,1260,5000
 1260 CALL IWRITE (ICONT, IDUMMY, 1)
      IF(IDUMMY)5000,260,5000
  260 CALL IWRITE (JXTIT, IDUMMY,11)
      IF(IDUMMY)5000,261,5000
  261 CALL IWRITE (JYTIT, IDUMMY,11)
      IF(IDUMMY)5000,265,5000
  265 IF(IT1)9269,9268,9269
 9268 CALL IWRITE (LTITLE, IDUMMY, 7)
      IF(IDUMMY)5000,9269,5000
 9269 IF(IT2)9271,9270,9271
 9270 CALL IWRITE (LTITLE(7), IDUMMY, 7)
      IF(IDUMMY)5000,9271,5000
 9271 CALL IWRITE (KAXIS, IDUMMY, 5)
      IF(IDUMMY)5000,9015,5000
 9015 IF(IPOINT)9020,270,9020
  270 CALL IWRITE (ICURV, IDUMMY, II)
      IF(IDUMMY)5000,9020,5000
 9020 IF(MODCURV - 1)272,272,9025
  272 IF(IGRID - 1)9025,273,9025
C         GENERATE GRID IF CALLED FOR.
  273 IX100 = ISIZEX*100
      IY100 = ISIZEY*100
      NEXT1 = IBOTMGN
      NEXT2 = LFTMGN + IX100
      JGRID(1) = 0
      DO 1274 J=1,19,2
      JGRID(J+1) = IPACK12 (LFTMGN, NEXT1, NEXT2, NEXT1)
      IF(NEXT1 - IBOTMGN - IY100)1273,1275,1275
 1273 NEXT1 = NEXT1 + 100
      JGRID(J+2) = IPACK12 (NEXT2, NEXT1, LFTMGN, NEXT1)
      IF(NEXT1 - IBOTMGN - IY100)1274,1276,1276
 1274 NEXT1 = NEXT1 + 100
 1275 JGRID(J+2) = IPACK12 (NEXT2, NEXT1, NEXT2, NEXT1)
```

```
 1276 JGRID(J+3) = ICURV3
      JGRID(J+4) = ICURV4
      CALL IWRITE(JGRID, IDUMMY, J+4)
      IF(IDUMMY)5000,1277,5000
 1277 NEXT1 = LFTMGN
      NEXT2 = IBOTMGN + IY100
      DO 1279 J=1,11,2
      JGRID(J+1) = IPACK12 (NEXT1, IBOTMGN, NEXT1, NEXT2)
      IF(NEXT1 - LFTMGN - IX100)1278,1280,1280
 1278 NEXT1 = NEXT1 + 100
      JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, IBOTMGN)
      IF(NEXT1 - LFTMGN - IX100)1279,1281,1281
 1279 NEXT1 = NEXT1 + 100
 1280 JGRID(J+2) = IPACK12 (NEXT1, NEXT2, NEXT1, NEXT2)
 1281 JGRID(J+3) = ICURV3
      JGRID(J+4) = ICURV4
      CALL IWRITE (JGRID, IDUMMY, J+4)
      IF(IDUMMY)5000,9025,5000
 9025 IF(IPOINT)9030,276,9030
C         GENERATE POINT PLOT RECORDS IF CALLED FOR.
 9030 IOUT = 0
      CALL IPACKL1 (LABEL, LABEL1, LABEL2)
      DO 9050 I=1,NUMPTS
      C1 = X(I)*SCX + SHIFTX
      C2 = Y(I)*SCY + SHIFTY
      IF(C1 - EXSIZE)9031,9031,9034
 9031 IF(C2 - YSIZE)9032,9032,9034
 9032 IF(C1 - SIZEX)9034,9033,9033
 9033 IF(C2 - SIZEY)9034,9035,9035
 9034 IOUT = IOUT +1
      GO TO 9050
 9035 IC1 = C1
      IC2 = C2
      GO TO (9036,9037,9038,9039,9040),IPOINT
C         GENERATE CROSS.
 9036 ICURV(2) = IPACK12 (IC1-5, IC2-5, IC1+5, IC2+5)
      ICURV(3) = IPACK12 (IC1  , IC2  , IC1-5, IC2+5)
      ICURV(4) = IPACK12 (IC1+5, IC2-5, IC1+5, IC2-5)
      GO TO 9041
C         GENERATE PLUS.
 9037 ICURV(2) = IPACK12 (IC1  , IC2-5, IC1  , IC2+5)
      ICURV(3) = IPACK12 (IC1  , IC2  , IC1-5, IC2  )
      ICURV(4) = IPACK12 (IC1+5, IC2  , IC1+5, IC2  )
      GO TO 9041
C         GENERATE SQUARE.
 9038 ICURV(2) = IPACK12 (IC1+4, IC2-4, IC1+4, IC2+4)
      ICURV(3) = IPACK12 (IC1-4, IC2+4, IC1-4, IC2-4)
      ICURV(4) = IPACK12 (IC1+4, IC2-4, IC1+4, IC2-4)
      GO TO 9041
C         GENERATE DIAMOND.
 9039 ICURV(2) = IPACK12 (IC1+5, IC2  , IC1  , IC2+5)
      ICURV(3) = IPACK12 (IC1-5, IC2  , IC1  , IC2-5)
      ICURV(4) = IPACK12 (IC1+5, IC2  , IC1+5, IC2  )
```

```
      GO TO 9041
C          GENERATE TRIANGLE.
 9040 ICURV(2) = IPACK12 (IC1+5, IC2-3, IC1  , IC2+6)
      ICURV(3) = IPACK12 (IC1-5, IC2-3, IC1+5, IC2-3)
      ICURV(4) = ICURV(3)
 9041 IF(I - NUMPTS)9043,9042,9043
 9042 ICURV(5) = LABEL2
      GO TO 9046
 9043 IF(I - 1)9045,9044,9045
 9044 ICURV(5) = LABEL1
      GO TO 9046
 9045 ICURV(5) = ICURV3
 9046 ICURV(6) = ICURV4
      CALL IWRITE (ICURV, IDUMMY, 6)
      IF(IDUMMY)5000,9050,5000
 9050 CONTINUE
      IF(IOUT)9048,276,9048
 9048 PRINT 9104, IOUT
 9104 FORMAT (/, 1X, I2, 29H POINT(S) WERE OFF THE GRAPH. ,/)
C          SET UP RETURN.
  276 IF(MODCURV)277,278,277
  277 IF(MODCURV - 3)279,278,279
  278 ITEST = 0
      PRINT 130,(JJTITLE(I),I=1,12)
  130 FORMAT (/, 19H GRAPH TITLED . .  ,6A8,/,19X,6A8,
     1           24H  . . HAS BEEN PLOTTED. ,/,1H0)
      IDUMMY = ITYP2(IDUMMY)
      IF(IDUMMY)5670,656,5670
  656 LAST = 0
      RETURN
  279 ITEST = 1
      IDUMMY = ICLOCK(IDUMMY)
      LAST = 0
      RETURN
C          THESE ARE THE NORMAL RETURNS.
C          NOW SET UP THE RETURN FOLLOWING A TAPE ERROR.
 5000 IF(MODCURV - 1)5001,5001,5002
 5001 IDUMMY = ITYPE1(IDUMMY)
      GO TO 247
 5002 PRINT 5100
 5100 FORMAT (/, 36H TAPE ERROR IN WRITING GRAPH OUTPUT. )
      IDUMMY = ITYPE1(IDUMMY)
      GO TO 1007
 5670 IDUMMY = ITYPE1(IDUMMY)
      END
C
C
      SUBROUTINE IREADY (IDUMMY)
C          SELECTS TAPE 8 (WILL LOOP UNTIL READY). WRITES EOF ON 8.
*          MACHINE LANGUAGE WILL NOT BE NECESSARY IN FORTRAN 62-3.
      LOC(IFIVE = 5).
       EXF (52041B)    EXF7 (52000B).   SELECT READ AND WAIT TAPE.
 1NEX  EXF7(00050B)    SLJ (1RDY).      EXIT ON CH 5 ACTIVE.
```

```
      EXF7(52000B)     SLJ (1NEX).        EXIT ON TAPE READY.
      LDA (IFIVE)      SAU (1BUF).        TERMINATE
 1BUF EXF5(N).                              BUFFER.
 1RDY EXF (52041B)     EXF7(52000B).      SELECT AND WAIT TAPE 8.
      ENA (0).                            CLEAR A.
      EXF (02000B)     EXF (52006B).      STOP CLOCK AND BACKSPACE 8.
      EXF7(52001B)     SLJ (1END).        EXIT IF NOT AT LOAD POINT.
     -EXF7(52000B).                       WAIT TAPE 8.
      EXF7(52007B)     SLJ (1EOF).        EXIT IF NO EOF.
      ENA (IDUMMY)     SAU (2BUF).        MOVE
      INA (1)          STA (IFIVE).         FORWARD
 2BUF EXF5(N)          EXF7(52000B).          OVER RECORD.
 1EOF ENA (0)          EXF7(00061B).      CLEAR A, WAIT CH 6.
      EXF (62041B)     EXF7(62000B).      SELECT WRITE AND WAIT TAPE.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE 8.
      EXF (62003B)     EXF7(62000B).      WRITE EOF AND WAIT.
      EXF7(62007B)     ENA(10).           EXIT ON NO END OF TAPE.
 1END STA (IDUMMY).
      END
C
      SUBROUTINE IWRITE(ISTART, IDUMMY, IWORDS)
C        WRITE RECORD OF IWORDS, STARTING WITH ISTART. PUT IDUMMY = 0
C         IF RECORD CORRECTLY WRITTEN, OTHERWISE SET NON-ZERO.
*         MACHINE LANGUAGE WILL NOT BE NECESSARY IN FORTRAN 62-3.
      LOC(ISIX = 6).
     -EXF7(00061B).                       WAIT CH 6.
      EXF (62041B)     EXF7(62000B).      SELECT WRITE, WAIT TAPE.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE 8.
      ENQ (111B).                         SET COUNTER.
 1AGN ENA (ISTART)     INA(1).            STARTING ADDRESS.
      SAL (1BUF)       ADD(IWORDS).       TERMINAL ADDRESS.
 1BUF STA(ISIX)        EXF6(N).           BUFFER OUT.
      ENA(0)           EXF7(62000B).      CLEAR A. WAIT TAPE 8.
      EXF7(62007B)     SLJ (1END).        EXIT IF NO END OF TAPE.
      EXF7(62003B)     SLJ (2AGN).        EXIT IF NO PARITY ERROR.
      EXF7(62004B)     SLJ (2END).        EXIT IF LENGTH ERROR.
 2AGN EXF (62006B)     EXF7(62000B).      BACKSPACE AND WAIT.
      QRS (3)          QJP1(1AGN).        TRY WRITE 3 TIMES.
 1END EXF (62003B)     ENA (10).          WRITE EOF, NON ZERO A.
 2END STA (IDUMMY).                       STORE RESPONSE.
      END
C
      FUNCTION ITYP2 (IDUMMY)
C         TYPE WORD GRAPH.
*         WILL NEED REWRITING IN FORTRAN 62-3.
      CON(LC = 57B, M1 = 4513123015050000B ).
      LOC(ITWO = 2).
     -EXF7(00061B).                       WAIT CH 6.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE 8.
      EXF (62003B)     EXF (01000B).      WRITE EOF. START CLOCK.
      ENA (0)          EXF7(62000B).      CLEAR A. WAIT TAPE 8.
      EXF7(62007B)     ENA (10).          EXIT IF NO END OF TAPE.
```

```
      STA (ITYP2).                        STORE RESPONSE.
     -EXF7(00021B).                       WAIT CH 2.
      EXF7(11141B)     SLJ (1TYP).        EXIT IF UPPER CASE.
      EXF (21100B)     ENA (LC+1).        TYPE
      STA (ITWO)       EXF2(LC).            LOWER CASE.
     -EXF7(00021B).                       WAIT CH 2.
 1TYP EXF (21100B)     ENA (M1+1).        TYPE
      STA (ITWO)       EXF2(M1).            GRAPH
     END
C
     FUNCTION ITYPE1 (IDUMMY)
C         REWIND TAPE 8, REQUEST NEW TAPE, AND WAIT TILL READY.
*         WILL NEED REWRITING IN FORTRAN 62-3.
     CON(LC = 57B, M1 = 4515112030242004B, M2 = 1103302204062031B,
    1    M3 = 0401301520043342B).
     RSV(MESS = 3).
     LOC(ITWO = 2).
     -EXF7(00061B).                       WAIT CH 6.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE 8.
      EXF (62003B)     EXF7(62000B).      WRITE EOF AND WAIT.
      EXF (62007B)     EXF7(00021B).      REWIND WITH INTERLOCK, WAIT CH 2
      EXF7(11141B)     SLJ (1TYP).        EXIT IF UPPER CASE.
      EXF (21100B)     ENA (LC+1).        TYPE
      STA (ITWO)       EXF2(LC).            LOWER CASE.
 1TYP LDA (M1)         STA (MESS).
      LDA (M2)         STA (MESS+1).
      LDA (M3)         STA (MESS+2).
     -EXF7(00021B).                       WAIT CH 2.
      EXF (21100B)     ENA (MESS+3).      TYPE
      STA (ITWO)       EXF2(MESS).          MESSAGE.
     -EXF7(00061B).                       WAIT CH 6.
     -EXF7(62000B).                       WAIT TAPE.
      EXF (62041B)     EXF7(62000B).      SELECT AND WAIT TAPE 8.
      EXF (01000B).                       START CLOCK.
     END
C
     FUNCTION ICODE (ANUMBER)
C         CODES ABSOLUTE VALUE OF A FLOATING POINT NUMBER (BETWEEN
C         1.0E-100 AND 1.0E+100) INTO 8-CHARACTER BCD WORD OF THE FORM
C         1.23E+45. ICODE = 8H0.00E+00 IF MAGNITUDE OUT OF RANGE.
*         CHECK AVAILABILITY OF LIBRARY FUNCTIONS IN FORTRAN 62-3.
     DIMENSION II(8)
     BNUMBER = ABSF(ANUMBER)
     IF(BNUMBER - 1.0E+100)7,6,6
   7 IF(BNUMBER - 1.0E-100)6,6,2
   6 ICODE = 8H0.00E+00
     RETURN
C         THIS IS ERROR EXIT.
   2 CALL SCALEIT (BNUMBER, ISCAL10, FACTOR, 3)
     ISIGNSC = XSIGNF(1,ISCAL10)
     ISCAL10 = XABSF(ISCAL10)
     IFACT = FACTOR*100.001
```

```
      II(8) = XMODF(ISCAL10,10)
      II(7)=ISCAL10/10
      IF(ISIGNSC)4,3,3
    3 II(6) = 8H       +
      GO TO 5
    4 II(6) = 8H       -
    5 II(5) = 8H        E
      II(4) = XMODF(IFACT,10)
      II(3) = (XMODF(IFACT,100))/10
      II(2) = 8H       .
      II(1) = IFACT/100
      CALL IPACK (II, IPACKED)
      ICODE = IPACKED
      RETURN
      END
C
      SUBROUTINE SCALEIT (ANUMBER,ISCAL10,FACTOR,MODE)
C         FINDS FACTOR (BETWEEN 1.0 AND 9.99...) AND SCALE OF 10 AS
C         DEFINED BY      ANUMBER = FACTOR*10.**ISCAL10.
C         MODE IS THE NUMBER OF SIGNIFICANT FIGURES REQUIRED. THIS MUST
C         BE BETWEEN 1 AND 10 OR IT WILL BE PUT EQUAL TO SIX.
*         CHECK AVAILABILITY OF LOG10F IN FORTRAN 62-3.
      ISCAL10=LOG10F(ANUMBER)
      FACTOR = ANUMBER/10.**ISCAL10
      IF(FACTOR - 0.1)1,2,2
    1 FACTOR = FACTOR*100.
      ISCAL10 = ISCAL10 - 2
      GO TO 8
    2 IF(FACTOR - 1.0)3,8,4
    3 FACTOR = FACTOR*10.
      ISCAL10 = ISCAL10 - 1
      GO TO 8
    4 IF(FACTOR - 100.0)6,5,5
    5 FACTOR = FACTOR/100.
      ISCAL10 = ISCAL10 + 2
      GO TO 8
    6 IF(FACTOR - 10.0)8,7,7
    7 FACTOR = FACTOR/10.
      ISCAL10 = ISCAL10 + 1
    8 IF(MODE)9,9,10
    9 MODE = 6
      GO TO 11
   10 IF(MODE - 10)11,11,9
   11 IFACTOR = FACTOR*10.**(MODE - 1) + 0.5
      FACTOR = IFACTOR
      FACTOR = FACTOR/10.**(MODE - 1)
      IF(FACTOR - 10.)13,12,12
   12 FACTOR = 1.
      ISCAL10 = ISCAL10 + 1
   13 RETURN
      END
C
      SUBROUTINE IPACK (II, IPACKED)
```

```
C          TAKES 8 SIX-BIT WORDS AND PACKS THEM LEFT TO RIGHT
C          IN IPACKED. IF WORD IS ZERO, 12B IS SUBSTITUTED.
*          CONVERT TO CODAP FOR FORTRAN 62-3.
      CON(IZERO = 12B).
       SIU1(ISAVE)      ENI1(8).
 1NEX  LDA1(II)         AJP1(2NEX).
       LDA (IZERO).
 2NEX  LRS (6)          INI1(-2).
       ISK1(-1)         SLJ (1NEX).
       STQ (IPACKED)    LIU1(ISAVE).
      END
C
      SUBROUTINE ISHIFT6 (ITITLE, LTITLE)
C          INSERTS 02B AHEAD OF 6-WORD TITLE RECORD.
*         WILL HAVE TO BE CONVERTED TO CODAP IN FORTRAN 62-3.
*          WATCH ARRAY INDEXING IN FORTRAN 62-3.
      CON(IBLANK = 2020202020202020B).
       SIU1(ISAVE)      ENI1(1).                SAVE INDEX, SET COUNT.
       ENA (2).                                 ENTER 02B.
 1NEX  LDQ1(ITITLE)     LLS (42).               PERFORM
       STA1(LTITLE)     LLS (6).                  SHIFTING.
       ISK1(6)          SLJ (1NEX).             CHECK IF COMPLETE.
       LDQ (IBLANK)     LLS (42).               COMPLETE LAST WORD.
       ENI1(7)          STA1(LTITLE).           STORE LAST.
       ENA (2).                                 REPEAT
 2NEX  LDQ1(ITITLE)     LLS (42).                 FOR
       INI1(1)          STA1(LTITLE).               SECOND
       INI1(-1)         LLS (6).                     TITLE
       ISK1(12)         SLJ (2NEX).                     LINE.
       LDQ (IBLANK)     LLS (42).
       ENI1(14)         STA1(LTITLE).
       LIU1(ISAVE).                             RESTORE INDEX.
      END
C
      FUNCTION IPACK12 (IONE,I2,I3,I4)
C          PACKS FOUR 12-BIT WORDS INTO ONE 48-BIT WORD.
*          WILL REQUIRE CONVERSION TO CODAP IN FORTRAN 62-3.
      LDA(IONE)      LDQ(I2).
      QLS(36)        LLS(12).
      LDQ(I3)        QLS(36).
      LLS(12)        LDQ(I4).
      QLS(36)        LLS(12).
      STA(IPACK12).
      END
C
      SUBROUTINE IPACKL1 (LABEL, LABEL1, LABEL2)
C          PACKS TWO 4-CHARACTER LABELS.
*          USE DECODE/ENCODE IN FORTRAN 62-3.
      CON(IFLAG = 37773777B).
       LDA (IFLAG)      LDQ (LABEL).
       LLS(24)          STA (LABEL1).
       LDA (IFLAG)      LLS(24).
       STA (LABEL2).
```

```
END

FUNCTION ICLOCK(IDUMMY)
 EXF (01000B).                    START CLOCK.
END
END
```

APPENDIX III

1. Derivations relating $\frac{\partial C}{\partial X}$ with $\frac{\partial d}{\partial X}$, and $\frac{\partial C}{\partial Y}$ with $\frac{\partial d}{\partial Y}$ [4].

The equation $$C = \frac{gT}{2\pi} \tanh\left(\frac{2\pi d}{TC}\right)$$

on rearrangement gives $$\frac{2\pi d}{TC} = \tanh^{-1}\left(\frac{2\pi C}{gT}\right).$$

The power series representation of the inverse hyperbolic tangent for values of $\left(\frac{2\pi C}{gT}\right)^2 < 1$ is

$$\tanh^{-1}\left(\frac{2\pi C}{gT}\right) = \frac{2\pi C}{gT} + \frac{1}{3}\left(\frac{2\pi C}{gT}\right)^3 + \frac{1}{5}\left(\frac{2\pi C}{gT}\right)^5 + \frac{1}{7}\left(\frac{2\pi C}{gT}\right)^7 + \ldots$$

so that to a good degree of accuracy it can be written

$$\frac{2\pi d}{TC} = \frac{2\pi C}{gT} + \frac{1}{3}\left(\frac{2\pi C}{gT}\right)^3 + \frac{1}{5}\left(\frac{2\pi C}{gT}\right)^5 + \frac{1}{7}\left(\frac{2\pi C}{gT}\right)^7$$

and rewriting

$$\frac{d}{c} = \frac{c}{g} + \frac{1}{3}\left(\frac{2\pi}{T}\right)^2\left(\frac{c}{g}\right)^3 + \frac{1}{5}\left(\frac{2\pi}{T}\right)^4\left(\frac{C}{g}\right)^5 + \frac{1}{7}\left(\frac{2\pi}{T}\right)^6\left(\frac{C}{g}\right)^7.$$

Then $$\frac{\partial d}{\partial C} = \frac{2c}{g} + \frac{4}{3}\left(\frac{2\pi}{T}\right)^2\left(\frac{C}{g}\right)^3 + \frac{6}{5}\left(\frac{2\pi}{T}\right)^4\left(\frac{C}{g}\right)^5 + \frac{8}{7}\left(\frac{2\pi}{T}\right)^6\left(\frac{C}{g}\right)^7$$

Since the depth may be considered as being a function of C $\left[d = F(C)\right]$, and C = G(X,Y), then by the chain rule

$$\frac{\partial d}{\partial X} = \frac{Dd}{DC}\,\frac{\partial C}{\partial X} \quad , \quad \frac{\partial d}{\partial Y} = \frac{Dd}{DC}\,\frac{\partial C}{\partial Y}$$

also since d is an explicit function of x and y

$$\frac{\partial^2 d}{\partial X^2} = \frac{\partial^2 C}{\partial X^2}\left[\frac{D^2 d}{DC^2} + \frac{Dd}{DC}\right] ,$$

$$\frac{\partial^2 d}{\partial X \partial Y} = \frac{\partial^2 C}{\partial X \partial Y}\left[\frac{D^2 d}{DC^2} + \frac{Dd}{DC}\right], \text{ and}$$

$$\frac{\partial^2 d}{\partial Y^2} = \frac{\partial^2 C}{\partial Y^2}\left[\frac{D^2 d}{DC^2} + \frac{Dd}{DC}\right].$$

2. Derivations of the partial derivatives of the depth function with respect to X and Y.

The derivatives may be computed directly from the equation for the quadric surface

$$d = A_1 + A_2 X + A_3 Y + A_4 XY + A_5 X^2 + A_6 Y^2$$

so that

$$\frac{\partial d}{\partial X} = A_2 + A_4 Y + 2A_5 X$$

$$\frac{\partial d}{\partial Y} = A_3 + A_4 X + 2A_6 Y$$

and

$$\frac{\partial^2 d}{\partial X^2} = 2A_5, \quad \frac{\partial^2 d}{\partial X \partial Y} = A_4, \quad \frac{\partial^2 d}{\partial Y^2} = 2A_6.$$

APPENDIX IV

Summary of refraction and shoaling relationships for intermediate depths [5] [6].

The following discussion applies to waves of small steepness where the deep-water wave height divided by the deep-water wave length is less than .005 ($H_0/L_0 < .005$). In all cases the subscript zero refers to deep-water parameters.

The wave velocity depends upon wave length and upon the depth of water:

$$C = \frac{gT}{2\pi} \tanh\left(\frac{2\pi d}{TC}\right)$$

where d is the depth, and T is the period of the wave.

Waves of a certain period curve as they approach the shore from deep water until, theoretically, they are perpendicular to the beach in zero depth of water. For any change in depth, Snell's law determines the curvature of the ray. It must intersect a contour at an angle determined by Snell's law for the successive changes in depth. The tangent to the wave ray must make an angle, α, with a perpendicular to the contour at the point where the ray intersects the contour. The ray must curve with the change in depth so that Snell's law is satisfied at a discrete set of points given by the intersection of the ray with a set of contours. As shown in Figure IV, the wave ray crosses the contour corresponding to the wave speed C_1. The tangent to the wave ray makes an angle α_1 with a line drawn perpendicular to the smoothed contour. Since the wave ray is continuously changing direction, it must make a new angle, α_2, with the perpendicular to the contour corresponding to the wave speed C_2, when it reaches that contour. The change in angle is $\Delta\alpha$. Then at the two contours corresponding to wave speeds, C_1 and C_2, Snell's law holds since

the wave crests intersect the contours at the correct angles. The important point is that the two ray tangents are connected by an arc of a circle which determines the exact path of the wave ray from point A to point B. The iteration procedure described in the text is used to arrive at this result.

The assumption behind the wave height calculation is that for steady state conditions energy does not flow across orthogonals and that none is destroyed by friction. Therefore, the power between orthogonals is assumed to remain constant. The mean wave energy per unit surface area equals:

$$E = \frac{1}{8} \rho g H^2$$

where ρ is the density of the water and H is the wave height. According to wave theory only a fraction of the wave energy is carried forward with the wave form at the speed C. Then the mean power transmitted between orthogonals equals:

$$P = n E C \, dl$$

where n is the fraction of energy carried forward and dl is distance between orthogonals. The numerical value of n approaches ½ in deep water and approaches one in shallow water. By equating the energy in the deep water to that in the shallow water, the ratio is formed:

$$\frac{E}{E_0} = \frac{1}{2} \frac{1}{n} \frac{1}{C/C_0} \frac{1}{dl/dl_0}$$

where the terms are defined above. This can be written:

$$\frac{H}{H_0} = \sqrt{\frac{1}{2} \frac{1}{n} \frac{1}{C/C_0}} \; K$$

where K is the refraction coefficient and is equal to $\sqrt{dl_0/dl}$. The term under the square root sign is termed the shoaling factor, H_s:

$$H_s = \sqrt{\frac{1}{2}\,\frac{1}{n}\,\frac{1}{C/C_0}}\ .$$

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
2.	Library U.S. Naval Postgraduate School Monterey, California 93940	2
3.	Professor Glenn H. Jung (Thesis Advisor) Dept. of Meteorology and Oceanography U.S. Naval Postgraduate School Monterey, California 93940	2
4.	LT David E. Stouppe, USN USS INDEPENDENCE (CVA-62) FPO, New York, New York 09501	2
5.	Department of Meteorology and Oceanography U.S. Naval Postgraduate School Monterey, California 93940	1
6.	Office of the U.S. Naval Weather Service U.S. Naval Station (Washington Navy Yard Annex) Washington, D.C. 20390	1
7.	Chief of Naval Operations OP-09B7 Navy Department Washington, D.C. 20350	1
8.	Officer in Charge Naval Weather Research Facility U.S. Naval Air Station, Bldg R-48 Norfolk, Virginia 23511	1
9.	Commanding Officer FWC/JTWC COMNAVMAR FPO San Francisco, Calif. 96630	1
10.	Commanding Officer U.S. Fleet Weather Central FPO San Francisco, Calif. 96610	1
11.	Commanding Officer Fleet Weather Central Navy Department Washington, D.C. 20390	1

12. Commanding Officer 1
Fleet Weather Central
U.S. Naval Air Station
Alameda, California 94501

13. Officer in Charge 1
Fleet Numerical Weather Facility
U.S. Naval Postgraduate School
Monterey, California 93940

14. U.S. Naval Oceanographic Office 1
Attn: Research Division
Washington, D.C. 20390

15. Director 1
Coast and Geodetic Survey
U.S. Department of Commerce
Attn: Office of Oceanography
Washington, D.C.

16. Office of Naval Research 1
Department of the Navy
Attn: Special Projects (Code 418)
Washington, D.C. 20360

17. Office of Naval Research 1
Department of the Navy
Attn: Geophysics Branch (Code 416)
Washington, D.C. 20360

18. Office of Naval Research 1
Department of the Navy
Attn: Director, Surface and Amphibious
Programs (Code 463)
Washington, D.C. 20360

19. Office of Naval Research 1
Department of the Navy
Attn: Senior Marine and Amphibious Warfare
Officer (Code 407M)

20. Council on Wave Research 1
Department of Civil Engineering
University of California
Berkeley, California

21. Coastal Engineering Research Center 1
Corps of Engineers, U S. Army
5201 Little Falls Road, N.W.
Washington, D.C. 20016

22. Chairman 1
Department of Meteorology and Oceanography
New York University
University Heights, Bronx
New York, New York

23. Chairman 1
Department of Meteorology and Oceanography
University of Hawaii
Honolulu, Hawaii

24. Chairman 1
Department of Oceanography
Oregon State University
Corvallis, Oregon 97331

25. Chairman 1
Department of Oceanography
Texam A&M University
College Station, Texas 77843

26. Director, Biological Laboratory 1
Bureau of Commercial Fisheries
U.S. Fish and Wildlife Service
450-B Jordan Hall
Stanford, California

27. Mission Bay Research Foundation 1
7730 Herschel Avenue
La Jolla, California
(Attn: Editor, Oceanic Coordinate Index)

28. Geophysical Institute 1
Tokyo University
Bunkyo-ku
Tokyo, Japan

29. Director 1
Pacific Oceanographic Group
Nanaimo, British Columbia
Canada

30. Ocean Research Institute 1
University of Tokyo
Tokyo, Japan

31. Secretary 1
Canadian Committee on Oceanography
Ottawa, Canada

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)*

U.S. Naval Postgraduate School
Monterey, California 93940

2a. REPORT SECURITY CLASSIFICATION

Unclassified

2b. GROUP

3. REPORT TITLE

OCEAN WAVE CREST AND RAY REFRACTION IN SHOALING WATER BY COMPUTER

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

Thesis, M.S., October 1966

5. AUTHOR(S) *(Last name, first name, initial)*

STOUPPE, David Edward

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
	64	6

8a. CONTRACT OR GRANT NO.

b. PROJECT NO.

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. AVAILABILITY/LIMITATION NOTICES

~~This document has been approved for public~~ release and sale; its distribution is unlimited.

[redacted]

11. SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY

13. ABSTRACT

The knowledge of wave refraction is important in many studies. The rapid and relatively easy gaining of this knowledge is made possible by the use of the modern high-speed digital computer. Large numbers of spectral periods and incoming directions are easily investigated, and immediate results are obtained by use of the plot of the wave crest refraction from the computer. This program presents the wave crest refraction pattern of numerous wave ray points rather than the single ray following technique. Its use is valuable in amphibious operation planning, and in other studies where the distribution of wave energy along the shore is desired for the many periods of the wave spectrum.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Wave refraction Numerical analysis						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ______________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ______________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ______________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

DD FORM 1 JAN 64 1473 (BACK)